HOCHSCHULE
MITTWEIDA
UNIVERSITY OF
APPLIED SCIENCES

MASTERARBEIT

Herr
Matthias Gay

Compressed Sensing and Sparse Reconstruction Algorithms

2013

Fakultät **Mathematik / Naturwissenschaften / Informatik**

MASTERARBEIT

Compressed Sensing and Sparse Reconstruction Algorithms

Autor:
Matthias Gay

Studiengang:
Diskrete und Computerorientierte Mathematik

Seminargruppe:
ZD10w1

Erstprüfer:
Prof. Dr. rer. nat. habil. Thomas Villmann

Zweitprüfer:
Prof. Dr.-Ing. Alexander Lampe

Mittweida, 2013

Bibliografische Angaben

Gay, Matthias: Compressed Sensing and Sparse Reconstruction Algorithms, 95 Seiten, 16 Abbildungen, 3 Tabellen, Hochschule Mittweida, Fakultät Mathematik / Naturwissenschaften / Informatik

Masterarbeit, 2013

Satz: LaTeX

Referat

Die vorliegende Arbeit befasst sich mit einem relativ jungen Gebiet innerhalb der angewandten Mathematik, dem *Compressed Sensing*. Darin geht es zum einen um die Frage, wie man einen dünn besetzten, hochdimensionalen Vektor durch lineare Projektionen in seiner Dimension reduzieren kann, ohne dass dabei Information verloren geht. Dieses 'Sensing'-Verfahren ist nicht-adaptiv, d.h. es soll für jeden dünn besetzten Vektor mit der gleichen Anzahl von null verschiedener Elemente gleichermaßen funktionieren und auch nicht während des Prozesses auf bereits berechnete Projektionen zurückgreifen. Zum anderen geht es um die Rekonstruktion des dünn besetzten Vektors aus diesen Projektionen, welche wesentlich weniger in der Zahl sind, als die Dimension des gesuchten Vektors. Das führt auf ein unterbestimmtes lineares Gleichungssystem, das zunächst unendlich viele Lösungen hat. Zusammen mit der Forderung der Dünnbesetztheit an die Lösung führt dies auf ein $\mathcal{NP}$-schweres kombinatorisches Optimierungsproblem. Die Theorie von Compressed Sensing bietet nun Antworten auf die Fragen, wann dieses Problem lösbar ist und wie man es mit effizienten Verfahren der Optimierung praktisch lösen kann.
Ziel dieser Arbeit ist es, einen einführenden Überblick in das *Sparse Reconstruction* Problem zu geben und eine Auswahl wichtiger theoretischer Resultate zu präsentieren. Dabei wird zum einen die konvexe Relaxation des $\mathcal{NP}$-schweren Problems betrachtet, welche mit Methoden der konvexen Optimierung effizient gelöst werden kann. Zum anderen werden schnelle Approximationsverfahren betrachtet, welche das kombinatorische Problem direkt approximativ lösen. Dabei wird sich auf eine Auswahl beschränkt, die keinen Anspruch auf Vollständigkeit erhebt, da es kurz nach Etablierung der Theorie eine wahre Explosion an Veröffentlichungen auf diesem Gebiet gab. Dennoch werden die wichtigsten Resultate und Ansätze präsentiert und durch simulative Berechnungen bestätigt und illustriert, wobei die präsentierten Verfahren zum großen Teil selbst implementiert wurden.

Contents

1 Introduction

1.1 Motivation

In the preceding decades, the digital revolution had a tremendous impact on the everyday life of almost everybody in the industrialized countries. Especially the field of communication and signal processing was affected, think for instance of mobile phones, digital cameras, digital television, or, even earlier, the compact disc. Nowadays we are dealing with high resolution television, and the average consumer can afford a high definition video camera which may actually be integrated in his or her smart phone. The technical part of this progress is driven by Moore's law, which postulates an exponential growth in computational power over time. The theoretical foundations reach back to the early 20th century and are essentially provided by the sampling theorem.

The sampling theorem was discovered independently by Whittaker (1915), Nyquist (1928), Kotelnikov (1933) and Shannon (1949), for a historical review see [63]. It says, for instance in the words of Claude Shannon in [77]:

> If a function $f(t)$ contains no frequencies higher than W cps[1], it is completely determined by giving its ordinates at a series of points spaced $\frac{1}{2W}$ seconds apart.

In other words, a bandlimited analogue signal can be reconstructed from uniformly spaced samples of it, if the sampling frequency is at least twice the largest frequency of the signal. This is referred to as *Nyquist sampling* or sampling at *Nyquist rate*. The reconstruction can be performed by sinc-interpolation, which is a very simple linear procedure and led to very efficient and cheap implementations during the development of electronic devices.

However, despite Moore's law, this classical approach reaches its limits at several points nowadays. On the one hand, Moore's law is not valid everywhere in electronics, and there are natural limits for the realizable sampling rate. On the other hand, an ever-growing resolution produces a tremendous amount of data and makes compression indispensable. A popular technique to accomplish this is transform coding. Here, one looks for a basis

[1] Cycle per second (cps) is an old unit for the frequency, it was later in 1960 replaced by the unit Hertz ($1\ \text{Hz} = 1\ \text{s}^{-1}$).

where the signal is *sparse* in, meaning that a large part of the coefficients of this representation are zero or almost zero, and only a very small part differs significantly from zero. Storing or transmitting only those few coefficients leads to a huge saving in data while preserving the relevant information. This is a standard procedure in image and video compression, such as JPEG or MPEG.

Thinking about this, one could ask if it is really necessary to sample at Nyquist rate, when after compression the same amount of information is condensed in a small fraction of the original data size. In other words: Is bandwidth the right measure for information?

It is indeed the sparsity thought that forges a bridge to the novel theory of *Compressed Sensing* (or *Compressive Sampling*). It provides an approach for sensing and reconstructing sparse signals, given much less measurements than the actual size of the signal, or from what was previously believed to be incomplete information. The theory was mainly established in the last decade by David Donoho, Emmanuel Candès, Justin Romberg, Terence Tao and others, see [15, 19, 36, 12] for instance.

The idea is to capture all relevant information by taking a small set of linear, non-adaptive measurements of a sparse signal, and it can be shown that the signal can be reconstructed exactly by a nonlinear procedure, namely by solving an optimization problem. Since the initial articles were published, there has been an explosion in the number of publications on that field. However, the first actual book on this topic [46] appeared as recently as in May 2012. We will give a short introduction to the problem modeling.

Suppose we have a vector $\mathbf{x}$ of length N with at most S non-zero entries, where $S \ll N$. The vector $\mathbf{x}$ is called *S-sparse* and the interpretation of N is the number of Nyquist samples. We want to reconstruct this vector from $M < N$ observations y_k in terms of linear projections (linear measurements), i.e. $y_k = \langle \varphi_k, \mathbf{x} \rangle$ for $k = 1, \ldots, M$. We can put the measurements into the *observation vector* $\mathbf{y}$ and state this as matrix equation

$$\mathbf{y} \;=\; \mathbf{\Phi}\,\mathbf{x}\,, \tag{1.1}$$

with the *measurement matrix* $\mathbf{\Phi}$ consisting of the measurement vectors φ_k^{H} as rows. For instance the situation may look like

$$\underbrace{\begin{bmatrix} \times \\ \times \\ \times \\ \times \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times & \times \end{bmatrix}}_{\mathbf{\Phi}} \cdot \underbrace{\begin{bmatrix} 0 \\ \times \\ 0 \\ \times \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{x}}, \tag{1.2}$$

where $N = 8$, $S = 2$, $M = 4$ and $\times$ denoting arbitrary entries. The problem of finding $\mathbf{x}$ is referred to as *sparse reconstruction*, which leads to solving an underdetermined system of linear equations under additional sparsity constraints. This turns out to be an $\mathcal{NP}$-hard combinatorial optimization problem.

The following questions concerning the existence of a unique solution and the ability to compute this solution were answered by the theory of Compressed Sensing in the last couple of years:

- How do the measurement vectors φ_k (and the resulting matrix $\mathbf{\Phi}$) have to look like to be able to reconstruct an arbitrary S-sparse vector $\mathbf{x}$?
- How many of them do we need? That is, what is a lower bound for M?
- How can we recover $\mathbf{x}$ in a computationally tractable manner?

Presenting answers to these questions will be a target of this thesis. Surprisingly it suffices to have a number of measurements M in the order of $S \log N$ to be able to recover $\mathbf{x}$, and the best thing that can happen is that the measurements are completely random. Moreover, reconstruction can be accomplished by solving a convex optimization problem, where efficient solvers exist for.

This thesis aims to give a rough overview and introduction to the huge research field of Compressed Sensing. We will present a selection of theoretical results that provide interesting insights in the capability of compression at the source of information and efficient reconstruction from compressed measurements. Due to the inherent structure of nature, this yields a huge potential for applications. In the author's opinion, Compressed Sensing will be part and parcel of future digital signal processing.

1.2 Organization of this Thesis

After introduction of some basic notation and definitions in the next clause, we will introduce the sparse reconstruction problem in Chapter 2. We begin with the original combinatorial optimization problem and present the idea to approach it: Relaxation in the ℓ_p sense. Especially we will introduce the ℓ_1-relaxation which is the convex relaxation of the original problem. We will also see how the standard least squares approach fits into the scheme of ℓ_p-relaxation, and why it is not an appropriate model in this case. Additionally, the model of noisy measurements is introduced and the related modified optimization problems are stated.

In Chapter 3 some theoretical results on reconstruction via ℓ_1-minimization are presented. The focus is on the measurement matrix $\mathbf{\Phi}$ and some properties of it to

guarantee reconstruction performance.

Chapter 4 presents alternatives to ℓ_1-minimization where the focus is set on greedy strategies. Here we present two important classes of greedy algorithms for sparse reconstruction and a selection of representatives of these classes.

Simulation results are presented in Chapter 5 in order to verify theoretical results and to compare different approaches. Here we restrict to a small selection of examples to make things visible. The work is concluded in Chapter 6.

1.3 Basic Definitions and Notation

The topic of this thesis employs several mathematical areas, such as linear algebra, functional analysis, probability theory and optimization. A basic knowledge is prerequisite, as well as the common notation. We will repeat some basic definitions and notation that will be frequently used in this thesis. A full list of abbreviations and symbols can be found at page 85ff.

We will primarily work with complex numbers $z \in \mathbb{C}$ and we denote the imaginary unit with $\mathrm{i} := \sqrt{-1}$, i.e. $z = a + \mathrm{i}\, b$ with $a, b \in \mathbb{R}$. The complex conjugate of z is denoted by $z^* = a - \mathrm{i}\, b$, the same notation holds for matrices and vectors. For vectors $\mathbf{x}$ and matrices $\mathbf{A}$ we denote with $\mathbf{x}^\mathsf{T}$ and $\mathbf{A}^\mathsf{T}$ the transposition, and with $\mathbf{x}^\mathsf{H}$ and $\mathbf{A}^\mathsf{H}$ the complex conjugate transposition.

We will need the following important terms from linear algebra.

Definition 1.1. An *orthonormal basis*, or *ortho-basis* $\boldsymbol{\Phi}$ of $\mathbb{C}^N$ is a set of N mutually orthonormal vectors that span $\mathbb{C}^N$, i.e. $\boldsymbol{\Phi} = \{\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_N\}$ with

$$\langle \boldsymbol{\varphi}_k, \boldsymbol{\varphi}_l \rangle \;=\; \delta_{k,l} \;=\; \begin{cases} 1 & ,\ k = l \\ 0 & ,\ \text{otherwise} \end{cases} \qquad \text{for all } \; k, l = 1, \dots, N\ . \tag{1.3}$$

The *canonical* or *spike* basis consists of the canonical unit vectors $\mathbf{e}_k \in \mathbb{C}^N$ with entries $(\mathbf{e}_k)_l = \delta_{k,l}$.
For a basis $\boldsymbol{\Phi}$ as above we will interchangeably use the representation as a matrix $\boldsymbol{\Phi} = [\, \boldsymbol{\varphi}_1 \,|\, \cdots \,|\, \boldsymbol{\varphi}_N \,]$.

In this thesis, if not stated otherwise, we will make use of the standard Euclidean inner product $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^\mathsf{H}\, \mathbf{v} = \sum_k u_k^*\, v_k$, although several results can be generalized to other inner products.

Definition 1.2. The *rank* of a matrix $\mathbf{A}$ is defined as the largest number of linearly independent rows or columns in $\mathbf{A}$.

Definition 1.3. The *kernel* or *null space* of an $M \times N$ matrix $\mathbf{A}$ over $\mathbb{C}$ is defined as

$$\ker(\mathbf{A}) \;:=\; \left\{ \mathbf{x} \in \mathbb{C}^N \mid \mathbf{A}\,\mathbf{x} = \mathbf{0} \right\} . \tag{1.4}$$

Definition 1.4. Let $\boldsymbol{V}$ be a vector space, $\boldsymbol{U} \subseteq \boldsymbol{V}$ a linear subspace, and $\mathbf{v} \in \boldsymbol{V}$. The by $\mathbf{v}$ *translated subspace* $\boldsymbol{U}$ is denoted

$$\boldsymbol{U} + \mathbf{v} \;:=\; \{ \mathbf{x} \in \boldsymbol{V} \mid \mathbf{x} = \mathbf{u} + \mathbf{v},\ \mathbf{u} \in \boldsymbol{U} \} . \tag{1.5}$$

Definition 1.5. Let $\boldsymbol{V}$ be a vector space over $\mathbb{C}$ and $\mathcal{S} \subset \boldsymbol{V}$ a finite set of vectors. The *linear span* of $\mathcal{S}$ is defined as the intersection of all subspaces of $\boldsymbol{V}$ that contain $\mathcal{S}$. It is given by the set of all linear combinations of vectors in $\mathcal{S}$

$$\operatorname{span}(\mathcal{S}) \;:=\; \left\{ \sum_{\mathbf{x} \in \mathcal{S}} \lambda_{\mathbf{x}}\, \mathbf{x} \;\middle|\; \lambda_{\mathbf{x}} \in \mathbb{C} \right\} . \tag{1.6}$$

Since we will work with sparse vectors we define the support set of a vector, and with this the concept of sparsity and approximate sparsity as follows.

Definition 1.6. Let $\mathbf{x} \in \mathbb{C}^N$. The *support* of $\mathbf{x}$ is the set of indices where $\mathbf{x}$ has non-zero entries, that is

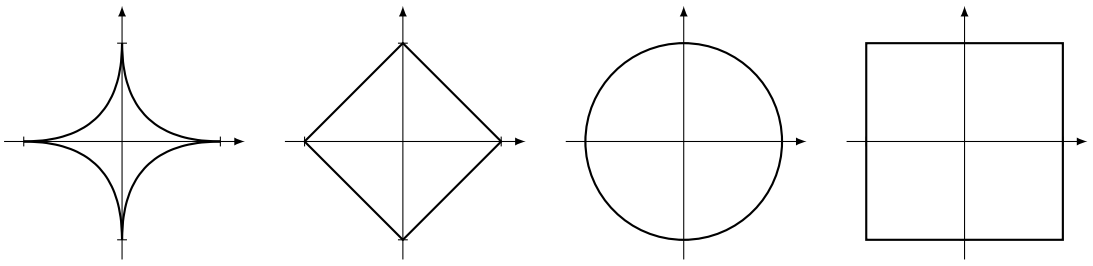
$$\operatorname{supp}(\mathbf{x}) \;:=\; \{ k \mid x_k \neq 0 \} . \tag{1.7}$$

Definition 1.7. A vector $\mathbf{x} \in \mathbb{C}^N$ is called *S-sparse* for some integer $S \leq N$, if it contains at most S non-zero entries, i.e. if $|\operatorname{supp}(\mathbf{x})| \leq S$.
The set of all S-sparse vectors in $\mathbb{C}^N$ is denoted by

$$\Sigma_S \;:=\; \left\{ \mathbf{x} \in \mathbb{C}^N \;\middle|\; |\operatorname{supp}(\mathbf{x})| \leq S \right\} . \tag{1.8}$$

We say that $\mathbf{x}$ is *approximately sparse* or *compressible* if the decreasingly sorted sequence of magnitudes of its entries decreases sufficiently fast (e.g. exponentially or by some power law). Given $\mathbf{x}$ and S we denote $\mathbf{x}_S$ the best S-sparse approximation to $\mathbf{x}$, obtained by setting all but the largest (in magnitude) S entries to zero.

A more precise definition of the concept of sparsity and compressibility will be given in Chapter 3.



(a) $p = \frac{1}{2}$ (b) $p = 1$ (c) $p = 2$ (d) $p \to \infty$

Figure 1.1: The unit ℓ_p-ball in $\mathbb{R}^2$ for different values of p.

Definition 1.8. Let $\mathbf{x} \in \mathbb{C}^N$ and $p \geq 1$. The *ℓ_p-norm* of $\mathbf{x}$ is defined as

$$\|\mathbf{x}\|_p := \left(\sum_{k=1}^{N} |x_k|^p \right)^{\frac{1}{p}} . \tag{1.9}$$

The extreme case $p \to \infty$ yields

$$\|\mathbf{x}\|_\infty := \lim_{p\to\infty} \|\mathbf{x}\|_p = \max_{k=1,\ldots,N} |x_k| . \tag{1.10}$$

We will also make use of the expression $\|\mathbf{x}\|_p$ for $0 < p < 1$, calculated via (1.9), which is not a norm but a quasinorm. A quasinorm satisfies the norm axioms, except for the triangle inequality, which is replaced by a relaxed version. Further, for the number of non-zero entries of a vector, we use the notation

$$\|\mathbf{x}\|_0 := |\operatorname{supp}(\mathbf{x})| . \tag{1.11}$$

This is motivated by the fact that $\lim_{p\to 0} \|\mathbf{x}\|_p^p = |\operatorname{supp}(\mathbf{x})|$, which can be shown easily from (1.9). Note that this is not a norm or quasinorm since generally positive homogeneity does not hold, i.e. $\|\alpha\,\mathbf{x}\|_0 \neq |\alpha|\,\|\mathbf{x}\|_0$ for arbitrary α.

Definition 1.9. For $p > 0$ and $R > 0$, the *ℓ_p-ball of radius R* is defined as the set of all vectors with ℓ_p-norm (or quasinorm) equal to R

$$B_p(R) := \left\{ \mathbf{x} \in \mathbb{C}^N \;\middle|\; \|\mathbf{x}\|_p = R \right\} . \tag{1.12}$$

The *unit ℓ_p-ball* has radius $R = 1$. The two-dimensional unit ℓ_p-ball (more exactly its real-valued analogue) is depicted in Figure 1.1 for different values of p. For $p \geq 1$ the ℓ_p-ball bounds a convex set, whereas for $p < 1$ it bounds a non-convex set and for $p \to 0$ it concentrates on the coordinate axes.

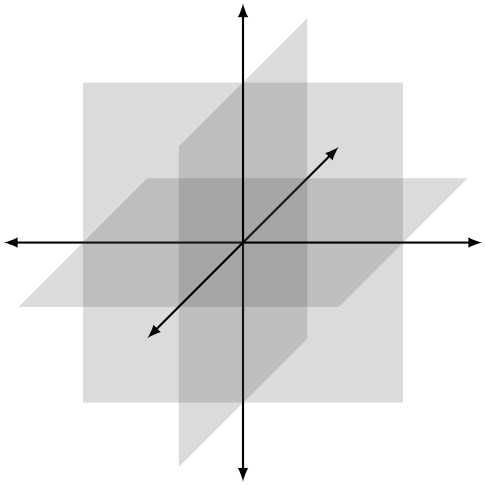


Figure 1.2: The ℓ_0-ball $B_0(2) = \Sigma_2$ in $\mathbb{R}^3$, which is the union of all two-dimensional Cartesian subspaces in $\mathbb{R}^3$.

We may also ask what the ℓ_0-ball may look like. By comparing (1.11) and (1.12) we first observe that the expression $B_0(R)$ only makes sense for integer values of R. The set $B_0(R)$ is the intersection of all R-dimensional Cartesian subspaces of $\mathbb{C}^N$, which are $\binom{N}{R}$ in number. A Cartesian subspace is spanned by a subset of the coordinate axes, i.e. with the canonical unit vectors $\mathbf{e}_i \in \mathbb{C}^N$. With this we can write

$$B_0(R) \;=\; \bigcup_{\substack{\mathcal{I}\subseteq\{1,\dots,N\}:\\ |\mathcal{I}|=R}} \operatorname{span}\left(\{\; \mathbf{e}_i \mid i \in \mathcal{I} \;\}\right). \tag{1.13}$$

This is a highly nonlinear set, for instance the 'unit' ℓ_0-ball (a wording that does not really make sense) $B_0(1)$ consists of the union of all coordinate axes. Note that $B_0(R)$ is just another description for Σ_R, defined in (1.8), since it equally denotes the set of all R-sparse vectors. For illustration the case $B_0(2)$ in $\mathbb{R}^3$ is depicted in Figure 1.2.

Later we will frequently use row and column selection of matrices and vectors. For this purpose we define the corresponding operators.

Definition 1.10. Let $\mathbf{A}$ be a matrix of size $M \times N$, and let $\mathcal{R} \subseteq \{1,\dots,M\}$ and $\mathcal{C} \subseteq \{1,\dots,N\}$ be subsets of the row indices and the column indices, respectively. The matrix $\mathbf{A}_{(\mathcal{R})}$ consists of the rows of $\mathbf{A}$, indexed by $\mathcal{R}$. We make the agreement that the indexing is in increasing order with respect to the corresponding index set. Analogously, the matrix $\mathbf{A}^{(\mathcal{C})}$ consists of the columns of $\mathbf{A}$, indexed by $\mathcal{C}$, and $\mathbf{A}^{(\mathcal{C})}_{(\mathcal{R})}$ consists of the rows of $\mathbf{A}$, indexed by $\mathcal{R}$ and the columns indexed by $\mathcal{C}$. The same notation holds for vectors, i.e. $\mathbf{x}_{(\mathcal{R})}$ is a vector of length $|\mathcal{R}|$ and consists of the elements of a vector $\mathbf{x}$ indexed by $\mathcal{R}$.

2 The Sparse Reconstruction Problem

2.1 A Combinatorial Optimization Problem

As initially introduced we consider the underdetermined system of linear equations

$$\mathbf{y} \;=\; \mathbf{\Phi}\,\mathbf{x}\,, \tag{2.1}$$

where $\mathbf{\Phi}$ is of size $M \times N$ with $M < N$, and we assume the unknown $\mathbf{x} \in \mathbb{C}^N$ to be S-sparse with $S \ll N$. Hence, from the uncountable number of solutions we want to pick one with the fewest number of non-zero entries. Using the 'ℓ_0-norm', defined in (1.11), this can be formulated as the following optimization problem, which is often referred to as the ℓ_0-minimization problem:

$$(P_0) \qquad \begin{aligned} &\text{Minimize} && \|\mathbf{x}\|_0 \\ &\text{subject to} && \mathbf{y} \;=\; \mathbf{\Phi}\,\mathbf{x} \end{aligned}$$

In other words, we want to find a subset $\mathcal{I} \subset \{1, \dots, N\}$ of the column indices of $\mathbf{\Phi}$ with $|\mathcal{I}| \leq S$ such that $\mathbf{y} = \mathbf{\Phi}^{(\mathcal{I})}\,\mathbf{x}_{(\mathcal{I})}$. This combinatorial optimization problem is $\mathcal{NP}$-hard, as shown for instance in [68] for real-valued instances. Solving this problem would require complete enumeration of all $\sum_{k=0}^{S} \binom{N}{k}$ possible subsets $\mathcal{I}$ of size up to S, which is computationally infeasible.

However, we are interested in the structure of $\mathbf{\Phi}$ that ensures unique solvability of (P_0) in principle. Especially of interest is the minimum number of rows of $\mathbf{\Phi}$, since this represents, from a sensing point of view, the number of linear measurements we have to take to be able to recover $\mathbf{x}$. From basic linear algebra we can give the following relatively simple statement.

Lemma 2.1. *If any subset of $2S$ columns of $\mathbf{\Phi}$ is linearly independent, then an S-sparse solution of (P_0) is unique. This requires $\mathbf{\Phi}$ to have at least $2S$ rows, i.e. $M \geq 2S$.*

Proof. Suppose we have two different S-sparse solutions $\mathbf{x} \neq \mathbf{x}'$ with $\mathbf{y} = \mathbf{\Phi}\mathbf{x} = \mathbf{\Phi}\mathbf{x}'$. Then $\mathbf{\Phi}\,(\mathbf{x} - \mathbf{x}') = \mathbf{0}$. Since the vector $\mathbf{x} - \mathbf{x}'$ is at most $2S$-sparse this implies that there exists a subset of at most $2S$ linearly dependent columns in $\mathbf{\Phi}$, in contradiction to the assumption of the lemma. □

One may ask if we can do better than $2S$. Clearly, the minimum number of rows has to be larger than S to avoid ambiguity. And indeed it has been shown in [4] that $S+1$ measurements suffice with probability one, if the matrix $\mathbf{\Phi}$ consists of i.i.d. Gaussian entries, independent of $\mathbf{x}$. We will recall the question for unique solvability and reconstruction properties of sensing matrices in the next chapter, for further detail reference is made to [4] and [41].

2.2 ℓ_p - Relaxation

A way to approach the ℓ_0-minimization problem is by relaxation in the ℓ_p sense. That means, straightforwardly we replace the term $\|\mathbf{x}\|_0$ by the ℓ_p-norm $\|\mathbf{x}\|_p$ for $p \geq 1$ or by the respective quasinorm for $0 < p < 1$, leading to the optimization problem:

$$\begin{array}{ll} \text{Minimize} & \|\mathbf{x}\|_p \\ \text{subject to} & \mathbf{y} = \mathbf{\Phi}\,\mathbf{x} \end{array}$$

The important case, which led to very interesting insights in the theory of Compressed Sensing, is the relaxation using $p = 1$, which is presented in the following.

2.2.1 Convex Relaxation: ℓ_1 - Minimization

Since sparsity as objective function in (P_0) is highly non-convex, we consider its convex relaxation in the ℓ_p sense – the ℓ_1-minimization problem:

$$(P_1) \qquad \begin{array}{ll} \text{Minimize} & \|\mathbf{x}\|_1 \\ \text{subject to} & \mathbf{y} = \mathbf{\Phi}\,\mathbf{x} \end{array}$$

In this sense, (P_1) is the closest convex problem to (P_0). Intuitively, this approach can be motivated by looking at the ℓ_p-ball for $p < 1$ and $p = 1$, see Figure 1.1, and the geometrical interpretation in Clause 2.2.3.

Since (P_1) is a convex problem it has a unique solution[1] and can be solved efficiently, e.g. by interior point methods. More precisely, it is a second-order cone program (SOCP), see [1] or [10] for instance. Moreover, in the case of real-valued $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{\Phi}$ it is actually a linear program.

[1] In fact, since (P_1) is not strictly convex, this is not generally true, but in this special kind of problem uniqueness is provided with high probability in dependence of the number M of rows of the matrix $\mathbf{\Phi}$, see Chapter 3.

The main question is under which conditions the solutions to (P_0) and (P_1) are equal to each other. Questions concerning these conditions are answered by Compressed Sensing theory, requiring the matrix $\mathbf{\Phi}$ to have specific properties which are outlined in Chapter 3. A key result is that with $M \gtrsim S \log N$ measurements reconstruction via ℓ_1-minimization is possible, and that this lower bound is sharp.

2.2.2 The Least Squares Solution

Solving an underdetermined linear system $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$ one can think of looking for the solution in the least squares sense, as it is done in linear regression. That is, we want to find an $\mathbf{x}$ that minimizes the expression

$$\|\mathbf{y} - \mathbf{\Phi}\,\mathbf{x}\|_2^2 \ .$$

The solution $\hat{\mathbf{x}}$ can be calculated explicitly[2] by means of

$$\hat{\mathbf{x}} = \mathbf{\Phi}^{+}\,\mathbf{y} \ , \tag{2.2}$$

where

$$\mathbf{\Phi}^{+} = \mathbf{\Phi}^{\mathsf{H}} \left(\mathbf{\Phi}\,\mathbf{\Phi}^{\mathsf{H}}\right)^{-1} \tag{2.3}$$

is the Moore-Penrose pseudoinverse[3] of $\mathbf{\Phi}$. To match our former notation we can transform the least squares reconstruction problem into the following equivalent optimization problem:

$$(P_2) \qquad \begin{array}{ll} \text{Minimize} & \|\mathbf{x}\|_2 \\ \text{subject to} & \mathbf{y} \ = \ \mathbf{\Phi}\,\mathbf{x} \end{array}$$

Solving the least squares problem seems to be a good initial guess since it leads to the solution with minimum energy, but it turns out that it leads to very poor results in sparse reconstruction. A reason is given geometrically in the next clause.

[2] Provided that $\mathbf{\Phi}$ has full rank, which we assume due to the specific application.
[3] See Appendix A.

2.2.3 Geometrical Interpretation

Given the linear equation $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$, where $\mathbf{\Phi}$ is of size $M \times N$ with $M < N$ and an arbitrary initial solution $\mathbf{x}_0$, we know that every vector in the translated null space

$$\mathcal{T} := \ker(\mathbf{\Phi}) + \mathbf{x}_0$$

is a solution thereof. Assuming $\mathbf{y}$ to be non-zero we have some non-zero minimum distance between $\mathcal{T}$ and the origin of coordinates, measurable with different ℓ_p-norms.

We can interpret the optimization problems (P_0), (P_1), (P_2) as follows: We search for an $\mathbf{x}$ in the intersection of $\mathcal{T}$ with an ℓ_p-ball of minimal radius, i.e.

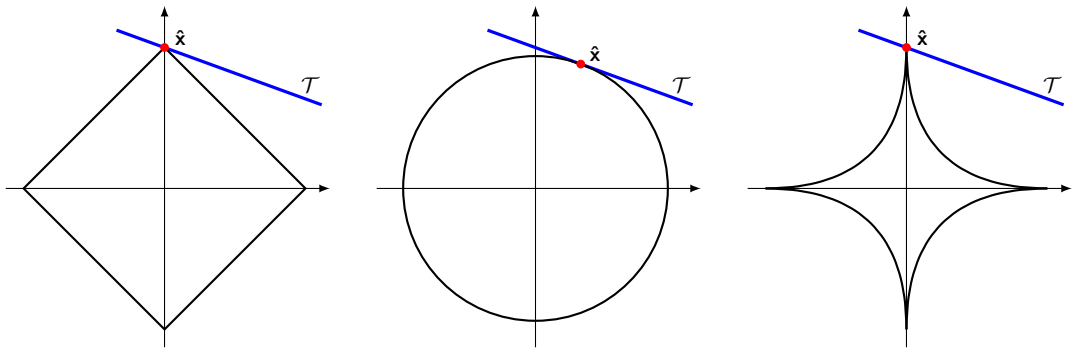
$$\mathbf{x} \in \inf_R \{\, \mathcal{M} = \mathcal{T} \cap B_p(R) \mid \mathcal{M} \neq \varnothing \,\} \,. \tag{2.4}$$

That is, $\mathbf{x} \in \mathcal{T} \cap B_p(R)$ for the smallest R such that $\mathcal{T} \cap B_p(R)$ is non-empty for the respective p of the corresponding optimization problem.

Begin with $R = 0$ and blow up an ℓ_p-ball until it touches the translated null space $\mathcal{T}$. The point where it first touches $\mathcal{T}$ is the point with minimal ℓ_p-norm and hence the solution to the respective optimization problem. If $p \geq 1$ the ℓ_p-ball is convex and the solution is unique.

Figure 2.1 shows the translated null space $\mathcal{T}$ together with the respective minimum-radius ℓ_p-balls for a very low-dimensional real-valued example. We clearly see that the ℓ_1-ball touches $\mathcal{T}$ at the coordinate axes; in other words the convex relaxation (P_1) decides for a sparse solution in $\mathcal{T}$. We also see why the least squares solution to (P_2) does not: The ℓ_2 solution decides for the element in $\mathcal{T}$ with minimum Euclidean distance from the origin. This might almost never be sparse, since for arbitrary problem instances we cannot expect $\mathcal{T}$ to be parallel to many coordinate axes.

The case $p = \frac{1}{2}$ is only depicted for illustration. The related relaxation of (P_0) – minimize $\|\mathbf{x}\|_{\frac{1}{2}}$ – may also find a sparse solution, but the problem is not convex which requires more sophisticated solution strategies. The approach will be briefly stated in Section 4.3.1. However, non-convex relaxation is not the main topic of this thesis.



(a) $p = 1$ (b) $p = 2$ (c) $p = \frac{1}{2}$

Figure 2.1: The translated null space and the respective ℓ_p solutions for different values of p.

2.3 The Noisy Reconstruction Problem

In preparation of practical applications we now consider the problem of noisy observations. According to the measurement equation (2.1), this can be modeled as

$$\mathbf{y} \;=\; \mathbf{\Phi}\,\mathbf{x} + \mathbf{n}\,, \tag{2.5}$$

where $\mathbf{n}$ is a stochastic or deterministic noise or error term. This noise term is often modeled as a vector with i.i.d. zero mean Gaussian entries, which corresponds with the common model of additive white Gaussian noise (AWGN). Note that model (2.5) captures both noisy observations and non-exactly sparse input vectors.

Due to the additional unknown term the reconstruction problem turns into an estimation problem. In analogy to (P_0) we state the sparse estimation problem as follows:

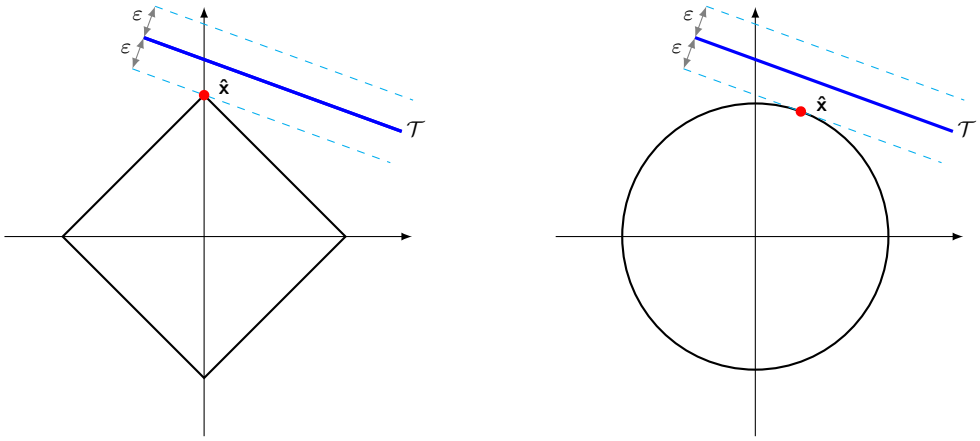
$$(P_0') \qquad \begin{aligned} &\text{Minimize} && \|\mathbf{x}\|_0 \\ &\text{subject to} && \|\mathbf{y} - \mathbf{\Phi}\,\mathbf{x}\|_2 \;\leq\; \varepsilon \end{aligned}$$

The parameter ε bounds the allowed deviation. In general it is required that $\varepsilon \leq \|\mathbf{n}\|_2$, see [37] for instance. This problem also includes the noiseless case (P_0) for $\mathbf{n} = \mathbf{0}$.

For reasonable ε this problem remains $\mathcal{NP}$-hard, so we may again consider the ℓ_1-relaxation, referred to as the noisy ℓ_1-minimization problem or Basis Pursuit denoising:

$$(P_1') \qquad \begin{aligned} &\text{Minimize} && \|\mathbf{x}\|_1 \\ &\text{subject to} && \|\mathbf{y} - \mathbf{\Phi}\,\mathbf{x}\|_2 \;\leq\; \varepsilon \end{aligned}$$

There are also different problem formulations involving the ℓ_1-norm, but we will stick with the one presented above. A selection of alternative ℓ_1-problems to solve the sparse



(a) $p = 1$ (b) $p = 2$

Figure 2.2: The translated null space with the allowed deviation ε and the respective ℓ_p solutions $p = 1, 2$.

estimation problem is presented in Appendix B.

For sake of completeness we also state the related least squares problem as ℓ_2-relaxation of problem (P_0'):

$$(P_2') \qquad \begin{aligned} &\text{Minimize} && \|\mathbf{x}\|_2 \\ &\text{subject to} && \|\mathbf{y} - \mathbf{\Phi}\,\mathbf{x}\|_2 \leq \varepsilon \end{aligned}$$

This problem is again equivalent to minimizing the expression $\|\mathbf{y} - \mathbf{\Phi}\,\mathbf{x}\|_2^2$ for a proper ε.

Geometrically, these ℓ_p-relaxations are similar to the noiseless ones, with the difference that a solution does not have to lay exactly in the translated null space $\mathcal{T}$, but up to a Euclidean distance of ε apart from $\mathcal{T}$. This is illustrated in Figure 2.2, again for the very low-dimensional example of a one-dimensional translated null space $\mathcal{T}$ in $\mathbb{R}^2$. Of course, there will be an estimation error, but from the illustration it can be motivated that the ℓ_1-solution again finds a sparse solution, and that the support is also detected correctly.

3 Compressed Sensing

The theory of Compressed Sensing mainly faces two topics. The first is the sensing process, which is where the name comes from. Main questions are on kind and number of measurements, required for capturing all information of an arbitrary S-sparse vector. The goal is a sensing process with as few as possible linear measurements in terms of inner products, which can be implemented very efficient in practical application setups.

The second topic is the reconstruction of the original information from these compressed measurements. This is exactly the sparse reconstruction problem from the previous chapter, where the answer was already presented: ℓ_1-relaxation.

In this chapter we present some interesting results concerning these two topics, namely incoherent sampling and other properties of sensing matrices, and the relation to the ability to reconstruct the signal via ℓ_1-minimization.

The idea is that the reconstruction process is allowed to be much more computationally expensive than the sensing process, but obviously we want to avoid solving an $\mathcal{NP}$-hard problem directly. Therefore we ask for conditions on the equality of solutions of the combinatorial ℓ_0-problem and its convex relaxation. More on the philosophy of Compressed Sensing will be concluded in Section 3.3.

3.1 Sparsity and Incoherence

3.1.1 Sparsity Models

In Definition 1.7 we defined the concept of sparsity and Σ_S as the set of all S-sparse vectors. Since exact sparsity rarely occurs in real-world applications, we start with introducing the concept of approximate sparsity or compressibility more precisely. At first, we specify the definition of a sufficiently fast decreasing sequence of entries.

Definition 3.1. Let $\mathbf{x} = (x_1, \dots, x_N)^\mathsf{T} \in \mathbb{C}^N$. Sort the coefficients of $\mathbf{x}$ such that $|x_{i_1}| \geq |x_{i_2}| \geq \dots \geq |x_{i_N}|$. The coefficients obey a *power law decay* if there exist

constants $C, r > 0$ such that

$$|x_{i_k}| \;\leq\; C\, k^{-r} \tag{3.1}$$

for all $k = 1, \ldots, N$.

With this it is possible to quantify the compressibility by the parameter r: The larger r is, the more compressible the vector $\mathbf{x}$ is. For a possible proper definition of compressibility we define the error, obtained by approximating a vector $\mathbf{x}$ by some S-sparse vector $\hat{\mathbf{x}}$.

Definition 3.2. Let $\mathbf{x} \in \mathbb{C}^N$, $S \leq N$ an integer, and $p > 0$. The *best S-term approximation error* is defined as

$$\sigma_S(\mathbf{x})_p \;:=\; \min_{\hat{\mathbf{x}} \in \Sigma_S} \|\mathbf{x} - \hat{\mathbf{x}}\|_p \;. \tag{3.2}$$

This error is easy to calculate, since it can easily be shown that the vector $\hat{\mathbf{x}} = \mathbf{x}_S$, obtained by setting all but the largest S entries in magnitude to zero (cf. page 5), yields the minimum of (3.2) for any $p > 0$.

Definition 3.3. Let $p \geq 1$ and $r > 0$. The vector $\mathbf{x} \in \mathbb{C}^N$ is called *p-compressible with constant C and rate r*, if

$$\sigma_S(\mathbf{x})_p \;\leq\; C\, S^{-r} \tag{3.3}$$

for any $S = 1, \ldots, N$.

A specific example of the connection between the power law decay and p-compressibility for the Euclidean case is the following [34].

Lemma 3.1. *For $p = 2$, the S-term approximation error is bounded by*

$$\sigma_S(\mathbf{x})_2 \;\leq\; C\, S^{-r} \tag{3.4}$$

if and only if the coefficients of $\mathbf{x}$ obey a power law decay with exponent $r - \frac{1}{2}$. That is,

$$|x_{i_k}| \;\leq\; \tilde{C}\, k^{-r+\frac{1}{2}} \;, \tag{3.5}$$

for the k-th largest entry x_{i_k} of $\mathbf{x}$, $k = 1, \ldots, N$.

For a given sparsity level S we define approximate sparsity by bounding the approximation error by a constant.

Definition 3.4. Let $S \leq N$ be an integer and $\delta \geq 0$. The vector $\mathbf{x} \in \mathbb{C}^N$ is called *δ-relatively S-sparse*, if

$$\sigma_S(\mathbf{x})_1 \;\leq\; \delta \,. \tag{3.6}$$

There are other ways to define the concepts of approximate sparsity and compressibility, but we will work with the ones defined above. Also, the concept of sparsity can be seen from a larger context of low-dimensional models. For instance, low-rank matrices fall into that topic, and the low-rank approximation of matrices is the analogue to sparse reconstruction in this case. Many results of the theory of Compressed Sensing can be generalized to other low-dimensional models. Further, one can define concepts of structured sparsity, which result from previous information of the sparse signals and their support, derived from practical applications. For a survey, see [46], [47] or [61].

3.1.2 Sparse Representation in a Proper Basis

Real-world signals or vectors are often not sparse in the first place, but they may be sparse when represented in a proper basis. Let $\mathbf{x} \in \mathbb{C}^N$ and $\boldsymbol{\Psi} = \{\psi_1, \dots, \psi_N\}$ an orthonormal basis of $\mathbb{C}^N$, such that

$$\mathbf{x} = \sum_{k=1}^{N} \psi_k \, c_k \, . \tag{3.7}$$

We can interpret the ortho-basis $\boldsymbol{\Psi}$ as a matrix $\mathbf{\Psi} = [\, \psi_1 \,|\, \cdots \,|\, \psi_N \,]$ with the basis vectors ψ_k as columns, and stack the coefficients into a vector $\mathbf{c} = (c_1, \dots, c_N)^{\mathsf{T}}$. This leads to the compact notation

$$\mathbf{x} = \mathbf{\Psi}\, \mathbf{c} \, . \tag{3.8}$$

If the coefficient vector $\mathbf{c}$ is (approximately) S-sparse we say that $\mathbf{x}$ is (approximately) S-sparse in the basis $\boldsymbol{\Psi}$ or under the unitary transform $\mathbf{\Psi}$. Conversely, we can obtain the coefficient vector by applying the 'sparsifying' inverse transform

$$\mathbf{c} = \mathbf{\Psi}^{\mathsf{H}} \mathbf{x} \, . \tag{3.9}$$

This is a reasonable model, since real-world data is almost never completely random but has an inherent structure. This structure often shapes out as sparsity in a proper basis, which is exploited in the following application.

Application: Transform compression

That sparsity is a reasonable assumption shows the application in transform compression. The idea is as follows:

- Transform the signal into a domain where it is (approximately) sparse.
- Define a threshold δ and set all coefficients with magnitude $< \delta$ to zero.
- Save the remaining coefficients (often accomplished by entropy-based source coding).

If the signal has very few significantly non-zero coefficients in transform domain, then this procedure is very efficient in terms of data reduction. It is for example implemented in the JPEG image compression standard [74, 82], where the sparsifying basis is a 2D discrete Cosine transform (DCT), the real-valued analogue of the discrete Fourier transform (DFT).[1] This is implemented in every digital camera to handle the huge amount of captured data. The more recent JPEG2000 standard [25] uses 2D wavelet transform, which provides even better results for natural images. We will illustrate the DCT-based compression in the following example.

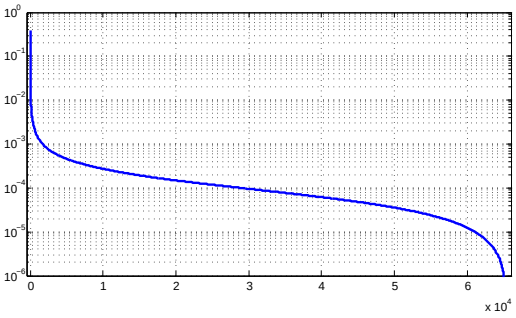
Example 3.1. Figure 3.1 (a) shows a 256×256 pixel gray-scale version of the famous Lenna test image[2]. Using 2D DCT we can observe that very few coefficients are very large, compared to the majority, see Figure 3.1 (b). The sub-figures (c) through (f) show the result of transform compression, keeping 25 %, 10 %, 5 %, and 2 % of the largest coefficients, respectively. The 25 % compression still provides good quality, even for sharper edges. Even when keeping only 2 % the content of the image is still visible. Note that this is not the exact way that JPEG works. JPEG divides the picture into blocks of size 8×8 pixels and performs DCT on those.

It should be mentioned that this is an example where sparsity can be exploited to significantly reduce the amount of data, but it differs fundamentally from the idea of Compressed Sensing. In transform compression all data has to be available to be compressed afterwards. Also, due to the thresholding strategy, this procedure is highly non-linear and it is important to know the exact positions of the kept non-zero elements. In Compressed Sensing we take linear measurements, and the number of these measurements is much less than the overall amount of captured data. The sensing process is non-adaptive and all we know is that the data is (approximately) sparse in some known basis and we roughly know the number of large coefficients, but not their position. We will specify that later after some theoretical considerations.

[1] The DCT is not a unitary transform, therefore one must work with the inverse transform instead of the conjugate transpose of the matrix. The results for unitary transforms translate to that case. Similar results can also be obtained by using the DFT instead, which is unitary.

[2] Image available for instance at `http://web.eecs.utk.edu/~qi/ece472-572/testimage.htm`. For the story of this test image see `http://www.lenna.org`.

(a) Original image



(b) Decreasingly sorted DCT coefficients on the logarithmic scale

(c) 25 %

(d) 10 %

(e) 5 %

(f) 2 %

Figure 3.1: The Lenna image and the sparse approximations using the indicated percentage of the largest DCT coefficients.

3.1.3 Incoherent Bases

In addition to a given orthonormal basis $\boldsymbol{\Psi} = \{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_N\}$ we will now consider another basis $\boldsymbol{\Phi} = \{\boldsymbol{\varphi}_1, \ldots, \boldsymbol{\varphi}_N\}$ which shall be used to measure samples of a sparse signal by means of linear projections $\langle \boldsymbol{\varphi}_k, \mathbf{c} \rangle$. We define the following important quantity.

Definition 3.5. For any given pair $(\boldsymbol{\Phi}, \boldsymbol{\Psi})$ of ortho-bases of size N, consisting of basis vectors as above, we define the *coherence* between $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ as

$$\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) \;:=\; \sqrt{N} \max_{k,l \in \{1,\ldots,N\}} |\langle \boldsymbol{\varphi}_k, \boldsymbol{\psi}_l \rangle| \,. \tag{3.10}$$

The coherence was first introduced as mutual coherence or incoherence in [42, 79] in an unscaled version. For the application it is an advantage to use the presented version scaled by $\sqrt{N}$. The value is bounded as follows.

Lemma 3.2. *For any pair of ortho-bases, it holds*

$$1 \;\leq\; \mu(\boldsymbol{\Phi}, \boldsymbol{\Psi}) \;\leq\; \sqrt{N} \,. \tag{3.11}$$

Proof. We restrict to the case $\langle \boldsymbol{\varphi}_k, \boldsymbol{\psi}_l \rangle = \boldsymbol{\varphi}_k^{\mathsf{H}} \boldsymbol{\psi}_l$ and denote $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ the matrices with the respective basis vectors as columns. Since both matrices are unitary, the

matrix $\mathbf{\Phi}^{\mathsf{H}}\mathbf{\Psi}$ with entries $\langle \varphi_k, \psi_l \rangle$ is also unitary. Due to normalized columns, the sum of absolute squares of entries of any unitary $N \times N$ matrix is N. Therefore, the maximal absolute value of an entry is at least $\frac{1}{\sqrt{N}}$, which with (3.10) yields the left inequality. The right inequality follows due to normalization $\|\varphi_k\|_2 = \|\psi_l\|_2 = 1$ directly from the Cauchy-Schwarz inequality $|\langle \varphi_k, \psi_l \rangle| \leq \|\varphi_k\|_2 \, \|\psi_l\|_2 = 1$. □

In Compressed Sensing we are interested in *in*coherent pairs, i.e. the coherence has a small value close to one and is independent of N.

Example 3.2. Let $\mathbf{x} \in \mathbb{C}^N$ and consider the following bases:

- $\mathbf{\Psi}$ – the canonical (spike) basis: $(\psi_k)_l = \delta_{k,l} = \begin{cases} 1 & , \text{ if } k = l \\ 0 & , \text{ otherwise} \end{cases}$
- $\mathbf{\Phi}$ – the Fourier basis: $(\varphi_k)_l = \frac{1}{\sqrt{N}} e^{-\mathrm{i} 2\pi \frac{k \cdot l}{N}}$

Observe that for the respective matrices we have $\mathbf{\Psi} = \mathbf{I}$ (the identity matrix) and therefore $\mathbf{x} = \mathbf{\Psi}\,\mathbf{c} = \mathbf{c}$, i.e. we interpret $\mathbf{x}$ as time domain representation; and $\mathbf{\Phi} = \mathbf{F}$ is the (normalized) $N \times N$ DFT matrix, i.e. $\mathbf{\Phi}\,\mathbf{x}$ is the respective frequency domain representation. It can be easily verified that the coherence of this pair is

$$\mu(\mathbf{\Phi}, \mathbf{\Psi}) = \mu(\mathbf{F}, \mathbf{I}) = 1\,, \tag{3.12}$$

which means minimum possible coherence, or 'maximal incoherence'.

Due to symmetry in the arguments of μ, we can switch the roles of $\mathbf{\Phi}$ and $\mathbf{\Psi}$ preserving the coherence. In the above example we know that if $\mathbf{x}$ is sparse in time domain then its frequency domain representation $\mathbf{\Phi}\,\mathbf{x}$ is usually dense and vice versa (except for periodic special cases), which is an important property of incoherent bases.

Example 3.3. The wavelet-noiselet pair:

- $\mathbf{\Phi}$ – any wavelet basis
- $\mathbf{\Psi}$ – noiselets (see Coifman et al. [27])

It has been shown that the coherence between Haar wavelets and noiselets is equal to $\sqrt{2}$, and the coherence between Daubechies wavelets and noiselets is ≤ 3 (cf. [21]). Noiselets are also maximally incoherent with spikes and incoherent with the Fourier basis.

The concept of bases can be generalized to the usage of *frames*, which provide more general and redundant representations. For an introduction, reference is made to [59, 60].

3.1.4 Incoherent Sampling and Noiseless Reconstruction

The role of incoherence will become clear when looking at the following results for taking linear measurements of a sparse vector as inner products with elements from an incoherent basis.

Recall the problem of signal recovery from under-sampled data. Suppose we are given a signal $\mathbf{x}$ which is sparse in the basis $\boldsymbol{\Psi}$ (matrix $\boldsymbol{\Psi}$), i.e. the coefficient vector $\mathbf{c} = \boldsymbol{\Psi}^{\mathsf{H}}\,\mathbf{x}$ is sparse. Further, we have M measurements $y_k = \langle \boldsymbol{\varphi}_k, \mathbf{c}\rangle = \langle \boldsymbol{\varphi}_k, \boldsymbol{\Psi}^{\mathsf{H}}\,\mathbf{x}\rangle$, with $\boldsymbol{\varphi}_k$ taken from the ortho-basis $\boldsymbol{\Phi}$ (i.e. columns of the matrix $\boldsymbol{\Phi}$), where $k \in \mathcal{M} \subset \{1, \ldots, N\}$ and $|\mathcal{M}| = M < N$. Now consider the optimization problem to try to recover the signal by ℓ_1-minimization:

$$\begin{array}{ll} \text{Minimize} & \left\|\boldsymbol{\Psi}^{\mathsf{H}}\,\mathbf{x}\right\|_1 \\ \text{subject to} & y_k \;=\; \langle \boldsymbol{\varphi}_k, \boldsymbol{\Psi}^{\mathsf{H}}\,\mathbf{x}\rangle \quad \text{for all} \quad k \in \mathcal{M} \end{array} \tag{3.13}$$

We put the measurements y_k into the observation vector $\mathbf{y} = (y_k)_{k\in\mathcal{M}}$ and create the measurement matrix $\tilde{\boldsymbol{\Phi}} = \left(\boldsymbol{\Phi}^{\mathsf{H}}\right)_{(\mathcal{M})}$, i.e. $\tilde{\boldsymbol{\Phi}}$ consists of the vectors $\boldsymbol{\varphi}_k^{\mathsf{H}}$ as *rows*, indexed by $k \in \mathcal{M}$. This yields the measurement equation

$$\mathbf{y} \;=\; \tilde{\boldsymbol{\Phi}}\,\mathbf{c}\,, \tag{3.14}$$

where $\mathbf{c}$ is sparse, and we can rewrite problem (3.13) to match the notation of the ℓ_1-relaxation for sparse reconstruction (P_1):

$$\begin{array}{ll} \text{Minimize} & \|\mathbf{c}\|_1 \\ \text{subject to} & \mathbf{y} \;=\; \tilde{\boldsymbol{\Phi}}\,\mathbf{c} \end{array} \tag{3.15}$$

The following theorem gives a statement about the solution $\hat{\mathbf{c}}$ of this problem and the recovered signal $\hat{\mathbf{x}} = \boldsymbol{\Psi}\,\hat{\mathbf{c}}$.

Theorem 3.1 (Candès, Romberg, 2006, [14])**.** *Let* $\mathbf{x}$ *be* S*-sparse in basis* $\boldsymbol{\Psi}$*. Select* M *measurements uniformly at random in the* $\boldsymbol{\Phi}$ *domain, so that*

$$M \;\geq\; C\,\mu^2\left(\boldsymbol{\Phi}, \boldsymbol{\Psi}\right)\,S\,\log N \tag{3.16}$$

for some given positive constant C*. Then the solution* $\hat{\mathbf{c}}$ *to the* ℓ_1*-minimization problem (3.15) reconstructs* $\mathbf{x}$ *exactly (i.e.* $\mathbf{x} = \boldsymbol{\Psi}\,\hat{\mathbf{c}}$*) with overwhelming probability.*

This is the first important result in this text related to Compressed Sensing, and at this point we see the importance of incoherent measurements: The lower the coherence,

the lower the number of measurements M we have to take to get exact recovery. With the wording *'overwhelming probability'* we mean that the probability of getting a wrong solution decreases exponentially when linearly increasing M.

Note that we can rewrite the measurements as

$$y_k \;=\; \langle \varphi_k, \mathbf{c} \rangle \;=\; \langle \varphi_k, \mathbf{\Psi}^{\mathsf{H}} \mathbf{x} \rangle \;=\; \langle \mathbf{\Psi}\, \varphi_k, \mathbf{x} \rangle \,, \tag{3.17}$$

that is, we can directly take linear measurements of $\mathbf{x}$ without previously applying a sparsifying transform, using the (pre-computed) measurement vectors

$$\tilde{\varphi}_k \;:=\; \mathbf{\Psi}\, \varphi_k \,. \tag{3.18}$$

The measurement matrix then consists of the vectors $\tilde{\varphi}_k^{\mathsf{H}}$ as rows and would be applied directly to $\mathbf{x}$. This leads to a huge saving in computation in the sensing process.

A Nonlinear Sampling Theorem

Of interest for many applications is the following, earlier discovered special case of Theorem 3.1, which can be considered as a new nonlinear sampling theorem (see Candès et al. [15]). Here we consider the frequency-time pair of ortho-bases $(\mathbf{\Phi}, \mathbf{\Psi}) = (\mathbf{F}, \mathbf{I})$, which we know to have maximal incoherence.

Theorem 3.2 (Romberg, Tao, 2004, [15])**.** *Let $\mathbf{x} \in \mathbb{C}^N$ be S-sparse in time domain. Select M frequency measurements uniformly at random, so that $M \geq C\, S \log N$ for some given positive constant C. Then minimizing ℓ_1 reconstructs $\mathbf{x}$ exactly with overwhelming probability.*

More precisely, if the constant is of the form $C = 22(\delta + 1)$, then the probability of wrong reconstruction is bounded by $\mathcal{O}(N^{-\delta})$, cf. [12].

Now consider Shannon's classical sampling theorem [77], which states that we can reconstruct a band-limited signal by sampling it at at least Nyquist rate. That is, if the signal has bandwidth B in frequency domain, and we sample in time domain uniformly spaced with sampling period $\leq \frac{1}{2B}$, then the signal can be reconstructed exactly using linear sinc-interpolation.
For large bandwidths this leads to very small sampling periods required for linear reconstruction. Now consider a signal with a very large bandwidth, but a relatively small number (say S) of non-zero frequency coefficients. Then conventional equidistant sampling at Nyquist rate would lead to a huge amount of data, which would actually be

required to reconstruct the original signal with conventional linear methods.
However, switching the roles of time and frequency, Theorem 3.2 tells us that we can get along with much less measurements, namely S times a logarithmic factor of the signal length, if we take the measurements in time domain uniformly at random. It uses the fact that we use the amount of *information* within the signal rather than its *bandwidth*.

3.2 Properties of Sensing Matrices

We will now take a look at some properties of the sensing matrix $\mathbf{\Phi}$ and their impact on the ability and quality of sparse reconstruction. In Section 3.1.4 we have already seen an important property of sensing matrices: Their rows should be incoherent with the basis in which the signal is sparse. From now on and without loss of generality, we will restrict again to the case where $\mathbf{x}$ is sparse itself, i.e. in the canonical basis, such that we can work with the sparse reconstruction problem and measurement equation formulations from Chapter 2.

3.2.1 Spark of a Matrix

To be able to reconstruct an S-sparse vector $\mathbf{x}$ uniquely from $\mathbf{\Phi}\,\mathbf{x}$, $\mathbf{\Phi}$ has to be an injective operator on the set of all S-sparse vectors, i.e.

$$\forall\, \mathbf{x}, \mathbf{x}' \in \Sigma_S : \;\; \mathbf{x} \neq \mathbf{x}' \;\Rightarrow\; \mathbf{\Phi}\,\mathbf{x} \neq \mathbf{\Phi}\,\mathbf{x}' \,. \tag{3.19}$$

To give statements on uniqueness of a solution in general, we define the following property [41].

Definition 3.6. The *spark* of a matrix $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ with $M < N$ is defined as the smallest number $\sigma = \mathrm{spark}(\mathbf{\Phi})$, such that there exists a subset of σ columns in $\mathbf{\Phi}$ that are linearly dependent.

A note on the connection between the spark the rank of a matrix: Although their definitions sound quite similar, there is a major difference. The rank is the *maximal* number of linearly *independent* columns and can be computed in $\mathcal{O}(N)$ steps. Computation of the spark on the other hand requires combinatorial search with $\mathcal{O}(2^N)$ steps. However, if $M < N$, we can state the bounds

$$2 \;\leq\; \mathrm{spark}(\mathbf{\Phi}) \;\leq\; \mathrm{rank}(\mathbf{\Phi}) + 1 \;\leq\; M + 1 \,. \tag{3.20}$$

A first result for reconstruction using the spark is the following, more precise formulation of Lemma 2.1.

Lemma 3.3. *An S-sparse solution of (P_0) is unique if and only if* $\operatorname{spark}(\mathbf{\Phi}) > 2S$.

The proof uses the same arguments as in Lemma 2.1, with the addition that both directions have to be shown. We omit it at this point.

3.2.2 Null Space Conditions

Using the spark we can make statements on the capability to recover exactly sparse vectors. We will now give some stronger conditions using the null space property of a matrix.

Definition 3.7. A matrix $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ satisfies the *null space property* (NSP) of order S, if

$$\left\|\mathbf{h}_{(\Lambda)}\right\|_1 \; < \; \frac{1}{2} \left\|\mathbf{h}\right\|_1 \tag{3.21}$$

holds for every $\mathbf{h} \in \ker(\mathbf{\Phi}) \setminus \{0\}$ and every $\Lambda \subset \{1, \ldots, N\}$ with $|\Lambda| \leq S$.

The condition (3.21) can be rewritten as

$$\sum_{k \in \Lambda} |h_k| \; < \; \sum_{k \notin \Lambda} |h_k| \,, \tag{3.22}$$

leading to the interpretation that the null space of $\mathbf{\Phi}$ contains no elements that are too concentrated, or approximately sparse. For instance, if $\mathbf{\Phi}$ satisfies a NSP of order S, then there is no S-sparse vector in $\ker(\mathbf{\Phi})$ satisfying the above inequality. The only element that would, if allowed, satisfy it with equality is the null vector.

The following theorem states the connection between NSP and uniqueness of a solution for exactly sparse vectors via ℓ_1-minimization.

Theorem 3.3. *An S-sparse solution of (P_1) is unique, if and only if the measurement matrix $\mathbf{\Phi}$ satisfies the NSP of order S.*

For a proof and some more interesting insights reference is made to the article of Cohen et al. [26], where one can also find results for performance guarantees for not exactly sparse vectors of the following form: Suppose $\Delta \colon \mathbb{C}^M \to \mathbb{C}^N$ is a reconstruction algorithm, that finds a solution $\hat{\mathbf{x}} = \Delta(\mathbf{y})$ to the sensing problem $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$, and we denote $(\mathbf{\Phi}, \Delta)$ as encoder-decoder pair. Can we give performance guarantees of the form

$$\left\| \mathbf{x} - \Delta(\mathbf{\Phi}\,\mathbf{x}) \right\|_1 \; \leq \; C\, \sigma_S(\mathbf{x})_1 \,, \tag{3.23}$$

and under which conditions? One result is the following.

Theorem 3.4. *Let M, N, $0 < S \leq N$ be integers, $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ a measurement matrix and $\Delta \colon \mathbb{C}^M \to \mathbb{C}^N$ a decoding algorithm. If $\mathbf{\Phi}$ has a NSP of order $2S$ with constant $\frac{C}{2}$, then there exists a decoder Δ such that $(\mathbf{\Phi}, \Delta)$ satisfies (3.23) with constant C. Conversely, if (3.23) holds for some decoder Δ, then the related $\mathbf{\Phi}$ has a NSP of order $2S$ with constant C.*

This follows from a more general result, which is stated and proven in [26]. It guarantees existence of an algorithm that provides a reconstruction with an error within the order of the best S-term approximation of $\mathbf{x}$ if some NSP is satisfied. Also, if $\mathbf{x}$ is S-sparse, then there exists an algorithm that reconstructs $\mathbf{x}$ exactly.

3.2.3 The Restricted Isometry Property

The following property, introduced by Candès and Tao in [18], appears to be the most important one to obtain theoretical results. Especially it can lead to performance results in the presence of noise.

Definition 3.8. Let $\mathbf{\Phi}$ be a matrix and let $S \geq 1$ be an integer. The *isometry constant* δ_S of $\mathbf{\Phi}$ is defined as the smallest number such that

$$(1 - \delta_S) \|\mathbf{x}\|_2^2 \;\leq\; \|\mathbf{\Phi}\,\mathbf{x}\|_2^2 \;\leq\; (1 + \delta_S) \|\mathbf{x}\|_2^2 \tag{3.24}$$

holds for every S-sparse vector $\mathbf{x} \in \Sigma_S$. Then $\mathbf{\Phi}$ is said to satisfy the *restricted isometry property (RIP)* of order S with constant δ_S.

If $\mathbf{\Phi}$ has a RIP with small δ_S, then any sub-matrix of S columns of $\mathbf{\Phi}$ acts nearly like an isometric operator. That means that the distances between S-sparse vectors $\mathbf{x} \in \Sigma_S$ are nearly preserved under transformation $\mathbf{\Phi}\,\mathbf{x}$.

Reconstruction Properties

There is a long list of results in Compressed Sensing theory, involving isometry constants to give statements on uniqueness, solvability, and robustness to noise and non-exact sparsity. The following theorems present a small selection thereof.

A first result on the unique solvability of the ℓ_0-problem using the RIP is the following, presented in [18].

Lemma 3.4. *If $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ satisfies a RIP with $\delta_{2S} < 1$, then an S-sparse solution of (P_0) is unique.*

Proof. Suppose there are two different S-sparse solutions $\mathbf{x} \neq \mathbf{x}'$ with $\mathbf{y} = \mathbf{\Phi x} = \mathbf{\Phi x}'$. Then $\mathbf{\Phi}(\mathbf{x} - \mathbf{x}') = \mathbf{0}$ and the vector $\mathbf{x} - \mathbf{x}'$ is at most $2S$-sparse. Taking the squared norm $\|\mathbf{\Phi}(\mathbf{x} - \mathbf{x}')\|_2^2 = 0$ and inserting it into (3.24) yields $(1 - \delta_{2S}) \|\mathbf{x} - \mathbf{x}'\|_2^2 = 0$. Since $\mathbf{x} \neq \mathbf{x}'$ this implies $\delta_{2S} = 1$, in contradiction to the assumption of the lemma. □

The following result, also from [18], gives RIP-based conditions for exact recovery if we solve the ℓ_1-relaxation of the sparse reconstruction problem.

Theorem 3.5 (Candès, Tao, 2005, [18])**.** *Assume that $\mathbf{x} \in \mathbb{C}^N$ is S-sparse with $S > 0$. We consider the measurement equation $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$ where $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ satisfies a RIP with $\delta_{2S} + \delta_{3S} < 1$. Then the solution $\hat{\mathbf{x}}$ of the ℓ_1-minimization problem (P_1) is exact, i.e. $\hat{\mathbf{x}} = \mathbf{x}$.*

This is an important result, since it tells us that the combinatorial problem (P_0) can be solved by solving its convex relaxation (P_1), provided that the sensing matrix meets the restricted isometry conditions. In the following theorem a very strong and general statement is given, since here the most realistic case is considered, where $\mathbf{x}$ is not exactly sparse and the measurements are also perturbed by noise.

Theorem 3.6 (Candès, Romberg, Tao, 2005, [16])**.** *Given a measurement model $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x} + \mathbf{n}$ as in (2.5) with $\|\mathbf{n}\|_2 \leq \varepsilon$. Let $\mathbf{x}_S$ be the best S-sparse approximation of $\mathbf{x}$, and the measurement matrix $\mathbf{\Phi}$ obeys a RIP with $\delta_{3S} + 3\delta_{4S} < 2$. Then for the solution $\hat{\mathbf{x}}$ of (P_1') it holds*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \;\leq\; C_0 \frac{\|\mathbf{x} - \mathbf{x}_S\|_1}{\sqrt{S}} \;+\; C_1\,\varepsilon \tag{3.25}$$

for some positive constants C_0 and C_1.

This is the most general result possible, since it gives a clear statement on robustness of ℓ_1-minimization for both non-sparsity and noise: The first term on the right-hand side of (3.25) gives a bound for the reconstruction error induced by non-exact sparsity, the second term bounds the error induced by the noise term. Especially for the effect of noise this result is best possible, since the error induced by that scales linearly with the noise floor. Also, Theorem 3.6 implies exact recovery with ℓ_1-minimization if the input vector is exactly S-sparse and measurements are noiseless.

A more recent result from Candès in [13] involves some simpler RIP condition, using only δ_{2S} with the same results as in Theorem 3.6.

Theorem 3.7 (Candès, 2008, [13]). *Given the measurement model, ε and $\mathbf{x}_S$ as in Theorem 3.6. If $\mathbf{\Phi}$ obeys a RIP of order $2S$ with $\delta_{2S} < \sqrt{2}-1$, then for the solution $\hat{\mathbf{x}}$ of (P_1') it holds*

$$\|\mathbf{x}-\hat{\mathbf{x}}\|_2 \;\leq\; C_0 \frac{\|\mathbf{x}-\mathbf{x}_S\|_1}{\sqrt{S}} \;+\; C_1\,\varepsilon \tag{3.26}$$

for some positive constants C_0 and C_1.

The dependency on δ_{2S} only is somehow more intuitive, since the sum of two S-sparse vectors is $2S$-sparse, which was already used to prove some simpler results above. For a proof of Theorem 3.7 we refer to [13]. Also, we refer to the original works for some more precise information about the constants involved. For some further results concerning reconstruction in presence of noise for different noise models, see [46].

We conclude the above results as follows.

Corollary 3.1.

(i) If $\delta_{2S} < 1$, then an S-sparse solution of the ℓ_0-problem (P_0) is unique.

(ii) If $\delta_{2S} < \sqrt{2}-1$, then S-sparse solutions of (P_0) and its convex relaxation (P_1) are equal.

(iii) If $\delta_{2S} < \sqrt{2}-1$, then (P_1) finds the best S-sparse approximation to an arbitrary input vector.

(iv) If $\delta_{2S} < \sqrt{2}-1$, then solving (P_1') is robust to noise and to non-exactly sparse inputs.

Measurement Bounds

An important question is again how many measurements M are required to obtain reconstruction results as above. In other words, how are the results for incoherent measurements in Section 3.1.3 related to the RIP? We will state only one result concerning this question.

Theorem 3.8 (Davenport, 2010, [31]). *If $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ satisfies a RIP of order $2S$ with $\delta_{2S} < 1$, then there exists $C > 0$ such that*

$$M \;\geq\; C\,S\,\log\frac{N}{S}\,. \tag{3.27}$$

The proof in the original work [31] led for instance to $C \approx 0.5$ if $\delta_{2S} = 0.25$. This was obtained without optimizing any constants or bounds. A more recent bound from [46] is that if $\delta_{2S} < 0.5$, then $C = \left(2\log\left(\sqrt{24}+1\right)\right)^{-1} \approx 0.28$. This rather small constant

leads to a small required number of measurements to form a matrix with the given RIP. Note that the statement is that if the RIP condition holds, then the lower bound for M holds. The more interesting question is how to construct matrices that reach that bound. We will face that question later on.

Connection to NSP

We will briefly state a connection between the RIP and the NSP. The following result shows that if a matrix satisfies a RIP, then it also satisfies some NSP. That is, RIP leads to more general results than NSP. The result is stated in the formulation of [61], for a proof see [46].

Theorem 3.9. *Let* $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ *satisfy a RIP of order* $2S$ *with* $\delta_{2S} < \sqrt{2} - 1$. *If*

$$\frac{\sqrt{2}\,\delta_{2S}}{1 - \left(1 + \sqrt{2}\right)\,\delta_{2S}} \;<\; \sqrt{\frac{S}{N}}\,, \tag{3.28}$$

then $\mathbf{\Phi}$ *satisfies the NSP of order* $2S$.

Downsides

Although the RIP is a mighty tool to obtain theoretical results, it has two major disadvantages. The first one is that for a given matrix $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ it is hard to determine the isometry constants. By definition of the RIP there are $\binom{N}{S}$ column sub-matrices to check for being nearly an isometry and to obtain the constant δ_S. So, one has to live with bounds, given by either other quantities or by computational approximation.

Secondly, there exists no deterministic procedure yet, that constructs matrices with a given RIP that reaches the bound of Theorem 3.8. Since the number of rows determines the number of required measurements, we are highly interested in reaching the lower bound in (3.27). So far, deterministically constructed sensing matrices have asymptotically far more rows than required by Theorem 3.8. For instance, the construction of DeVore in [35] requires $M \gtrsim S^2 \log N$ rows, and that one in [56] requires $M \gtrsim S\,N^{\alpha}$ for some constant α.

Fortunately, some classes of random matrices have been shown to reach the bounds with very high probability. In [3] this has been shown for a whole class of random matrices whose probability distributions satisfy a specific concentration inequality. We will state only the special cases of the Gaussian and the Bernoulli distribution here.

Theorem 3.10 (Corollary of Theorem 5.2 in [3]). *Let* $0 < \delta < 1$, $\mathbf{\Phi}_1$ *an* $M \times N$ *matrix with entries* $\pm\frac{1}{\sqrt{M}}$ *independently with equal probability, and* $\mathbf{\Phi}_2$ *an* $M \times N$ *matrix with i.i.d. zero-mean Gaussian entries with variance* $\frac{1}{M}$. *Then there exist positive constants* C_0 *and* C_1 *such that if*

$$M \;\geq\; C_0\, S \log \frac{N}{S} \tag{3.29}$$

the matrices $\mathbf{\Phi}_1$ *and* $\mathbf{\Phi}_2$ *satisfy the RIP of order* S *with constant* δ *with probability at least* $1 - 2\, e^{-C_1 M}$.

This is somehow related to the condition of incoherent sensing, discussed in Section 3.1.4. Imagine a set of M vectors whose entries are drawn i.i.d. from a Gaussian distribution with variance $\frac{1}{N}$ (a slightly different model than in the theorem, but basically just a scaling). For large N, it can be shown that these vectors are nearly orthonormal and also incoherent to any structured basis. If we use these as sensing vectors, we can apply Theorem 3.1 and obtain a similar result to the above theorem. Indeed, random projections lead to the best known results in the theory of Compressed Sensing.

Similar, but slightly weaker results hold for matrices that consist of M rows of the $N \times N$ DFT matrix $\mathbf{F}$, selected uniformly at random without replacement, cf. [76]. This is also related to the results of Theorem 3.2.

3.2.4 Mutual Coherence

An easy to compute property of sensing matrices, that still provides some insights, is the following.

Definition 3.9. Let $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ be a matrix with columns denoted by $\boldsymbol{\varphi}_k$, $k = 1, \ldots, N$. The *mutual coherence* of $\mathbf{\Phi}$ is defined as

$$\mu(\mathbf{\Phi}) \;:=\; \max_{k \neq l} \frac{|\langle \boldsymbol{\varphi}_k, \boldsymbol{\varphi}_l \rangle|}{\|\boldsymbol{\varphi}_k\|_2 \, \|\boldsymbol{\varphi}_l\|_2}\,. \tag{3.30}$$

This property, for instance defined in [40], is somehow related to the coherence between two bases, defined in Definition 3.5 at page 19. We can interpret $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$ as a *dictionary* with N elements $\boldsymbol{\varphi}_k$ of dimension M. The sparse reconstruction problem can be interpreted as representing the observation vector $\mathbf{y}$ as a linear combination of as few as possible dictionary elements. A dictionary is the analogue to a basis, with the difference that a dictionary can be redundant. For instance it can consist of the elements of two ortho-bases, say $\mathbf{\Phi}$ and $\mathbf{\Psi}$. Then the coherence of this dictionary is exactly the coherence between these two bases, except for the scaling factor $\sqrt{N}$. However, note

that in Section 3.1.4 we constructed the measurement matrix from basis elements as *rows*, whereas here we measure the coherence using its *columns*. The dictionary point of view will be taken up in the next chapter.

Trivially, $0 \leq \mu(\mathbf{\Phi}) \leq 1$ holds; 0 for unitary (square) matrices and 1 if there are two collinear columns in $\mathbf{\Phi}$. A non-trivial lower bound is

$$\mu(\mathbf{\Phi}) \;\geq\; \sqrt{\frac{N-M}{M(N-1)}}\,, \tag{3.31}$$

which is known as the Welch bound [84]. Observe that if N is much larger than M we obtain approximately $\mu(\mathbf{\Phi}) \geq \frac{1}{\sqrt{M}}$.

Reconstruction Properties

There are several results using the mutual coherence and variants thereof for statements on sparse reconstruction. We only state the following, cf. [45].

Lemma 3.5. *Let $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ and $\hat{\mathbf{x}}$ an S-sparse solution of (P_0) with*

$$S \;<\; \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{\Phi})}\right)\,. \tag{3.32}$$

Then $\hat{\mathbf{x}}$ is the unique solution to both (P_0) and (P_1).

This yields the general rule that the lower the mutual coherence is, the better are the sparse reconstruction properties for ℓ_1-minimization.

Connections to Spark and RIP

The mutual coherence is related to the spark of a matrix in the following way.

Lemma 3.6. *For any matrix $\mathbf{\Phi}$ it holds*

$$\operatorname{spark}(\mathbf{\Phi}) \;\geq\; 1 + \frac{1}{\mu(\mathbf{\Phi})}\,. \tag{3.33}$$

The proof is quite short and not too complicated, and uses Geršgorin's circle theorem [52]. However, we omit it here and refer to [46]. Finally, a relation between the mutual coherence and the RIP is as follows.

Lemma 3.7. *Let $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ have normalized columns. If $S < \frac{1}{\mu(\mathbf{\Phi})}$, then $\mathbf{\Phi}$ satisfies a RIP of order S with constant $\delta_S = (S-1)\,\mu(\mathbf{\Phi})$.*

The proof is similar to that of Lemma 3.6, and we also refer to [46].

Note that, since both spark and RIP are hard to calculate, the mutual coherence can be used to obtain bounds on those quantities. It can further be used to construct sensing matrices with the objective of minimizing the mutual coherence, which is far easier than optimizing the other quantities.

3.3 The Compressed Sensing Paradigm

As already stated, Compressed Sensing provides a new theory for data acquisition and reconstruction, if there is a sparse (or more generally speaking, a low-dimensional) representation of the data. The idea is to directly capture data in a compressed form, instead of unnecessarily capturing huge amounts of data and compressing it afterwards.

In Figure 3.2 the difference between Compressed Sensing and classical signal acquisition with transform compression coding is illustrated. The classical approach suffers from three inefficiencies at the source. First, to meet the classical sampling theorem, a large number N of samples has to be captured, even if the information within is much less. Second, since transform compression is adaptive and data-dependent, all N transform domain coefficients have to be calculated, even though only a few of them (say S) are stored and the remaining are discarded. And third, there is an additional source coding step, which also requires some computational power.

The Compressed Sensing approach on the other hand captures only a number of $\mathcal{O}(S \log N)$ linear measurements. These are non-adaptive, resulting in saving both computational power and amount of data to handle at the source. Clearly, due to some more sophisticated reconstruction methods, the computational requirements at the decoder are larger, but this is exactly what can be of advantage for several applications: moving computational complexity from the source of information to the decoder.

To exemplify this, consider the operating principle of a digital camera. The related sampling step in Figure 3.2 (a) is performed by the image sensor, which captures for example a 10 megapixel image. That means $N = 10^7$. Since this is too much to store directly, the camera saves the image for instance in JPEG format. That means DCT plus subsequent source coding, resulting in an enormous data reduction.

Actually, an image capturing method based on Compressed Sensing was already implemented in 2006 – the single pixel camera at Rice University.[3] The setup is described in [43] and is a straightforward implementation of Figure 3.2 (b). The inner product with

[3] See `http://dsp.rice.edu/cscamera`.

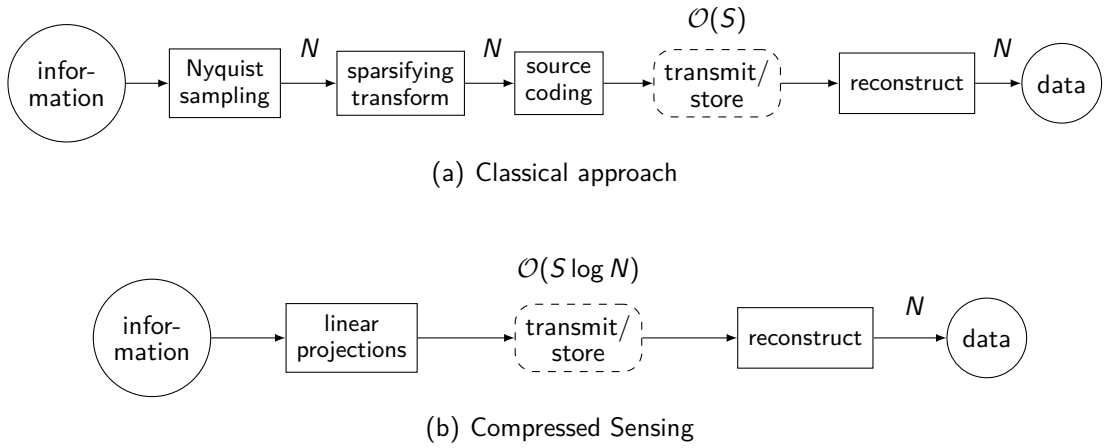


Figure 3.2: Schematic view of (a) conventional data acquisition with transform domain compression vs. (b) Compressed Sensing.

some incoherent basis element is implemented by a digital micro-mirror array which can reflect the image superimposed by some random pattern. The image sensor is a single photo diode that captures only the amount of reflected light. Capturing the value for M different random patterns yields the measurement vector, from which the image is reconstructed by solving a convex optimization problem

Of course, since image sensors of digital cameras are cheap, this may not be the target application, but the experiments show that it already works. And the concept can be applied for capturing images at wavelengths where it is much more expensive to built megapixel sensors. In the Compressed Sensing setup one only has to change the photo diode.

The results in Compressed Sensing theory also led to major insights in the field of sparse reconstruction, apart from the sensing process itself. In many applications inverse problems with sparsity constraints appear, where the origin of the data inherently turns out to result from what can be seen as an incoherent sensing process. An example is magnetic resonance imaging (MRI), where earlier results where obtained. Here the technology is was not advanced enough to meet the requirements of the sampling theorem, with the result that least squares led to poor reconstruction results. Candès and Tao (see [15], an earlier version of this paper is from 2004) observed by accident that reconstruction from incomplete frequency information is indeed possible, when they tried out ℓ_1-minimization instead of ℓ_2 (least squares).

Another application is the estimation of mobile radio channels. In orthogonal frequency-division multiplexing (OFDM, see [81]) systems the demodulation is performed by a DFT, and at specific positions, referred to as pilots, the transmitted signal is known

to the receiver. On these pilot positions noisy estimates of the channel frequency response can be calculated, which are noisy frequency measurements of the channel impulse response. For physical reasons, this impulse response is sparse and it can be reconstructed from the incomplete frequency information obtained from the pilots by using insights from Compressed Sensing theory, here in particular Theorem 3.2. Conventional methods rarely exploit the inherent sparsity of this model.

4 Reconstruction Algorithms

The theoretical results in the previous chapter lead to the conclusion that, if the measurement matrix $\mathbf{\Phi}$ is chosen properly, the $\mathcal{NP}$-hard problems (P_0) and (P_0') can be solved by solving their convex relaxations (P_1) and (P_1') instead. In this chapter we will discuss some basic approaches to efficiently solve the sparse reconstruction problem, where we will not restrict to ℓ_1-minimization. As it turns out, ℓ_1-relaxation is only one out of several ways to approach the sparse reconstruction problem. A large part of this chapter will be focused on fast iterative algorithms that search directly for a sparse solution. But to start with we want to discuss some aspects of the convex relaxation.

4.1 Basis Pursuit

Basis Pursuit (BP) is not a method or algorithm, but a designation for the ℓ_1-optimization problem (P_1). It was introduced in [24] in the context of representing a signal by a small number of elements φ_k of an overcomplete dictionary. As we have already seen in Section 3.2.4, this is an equivalent viewpoint on the sparse reconstruction problem, if we interpret the dictionary elements φ_k as columns of the sensing matrix $\mathbf{\Phi}$. The dictionary learning point of view leads to new insights and is of special interest for the greedy algorithms in the next section.

Since the ℓ_1-problem is convex, there exist efficient solvers, and the increasing interest in convex optimization led to a huge amount of research in this field, cf. [10]. The approaches for solving the BP problem are mainly using interior point methods, often specially designed for this kind of second-order cone program. Also we want to recall that the special case of a real-valued and noiseless ℓ_1-problem is actually a linear program that can in general be solved using the Simplex algorithm.

We want to emphasize the following fact for the ℓ_1-relaxation: Once we have formulated the ℓ_1-minimization problem and the requirements are met, we can use whatever solver comes in mind for convex optimization problems. The solution will turn out to be sparse by itself, although we do not explicitly search for or force sparsity. This is one of the major results of Compressed Sensing theory. The explicit search for sparse solutions will be topic of the following sections.

However, for real-time applications and large-scale problems, convex optimization requires still a huge amount of computational power. In this case, sparse approximation algorithms with fast heuristics should be preferred. This is where greedy algorithms appear on the scene.

4.2 Greedy Approaches

The greedy approach for sparse reconstruction does not follow that of convex relaxation but rather tries to calculate an approximation to the ℓ_0-problem iteratively. Due to the variety of applicable approaches and heuristics from other fields, there is a huge amount of research in this field, which led to some very good and fast algorithms. Some of these algorithms have performance guarantees similar or close to those for convex relaxation techniques.

In the following we will present two approaches to apply the greedy strategy to the problem of sparse reconstruction. The first one leads to the class of directional pursuits, which construct the support and the solution on this support step by step by directional updates. The second one uses thresholding strategies, where in each iteration some estimate is calculated and then forced to be sparse afterwards.

For further reading we refer for instance to [78, 80] and the references to the considered algorithms in this section, as well as Chapter 8 of the book [46].

4.2.1 Directional Pursuits

The class of directional pursuits provides algorithms that successively refine a current solution by going into some update direction, calculated from the previous solution and the correlation of the previous residual and the columns φ_k of the measurement matrix $\mathbf{\Phi} \in \mathbb{C}^{M \times N}$. Here comes why the dictionary point of view is suitable, since we look for a dictionary element that has the largest impact on the current residual.

The generic version of the directional pursuit framework, as presented for instance in [8, 33], is stated in Algorithm 1. Here are some additional notes on the specific steps, and also an introduction to the notation used.

We begin with an empty support set $\mathcal{I}$, belonging to the initial solution vector $\mathbf{x} = \mathbf{0}$, and we initialize the residual $\mathbf{r}$ with the observation vector $\mathbf{y}$ since the meaning of the residual is $\mathbf{r} = \mathbf{y} - \mathbf{\Phi}\,\mathbf{x}$ for the current estimate of $\mathbf{x}$. The vector $\mathbf{g}$ in line 3 can be interpreted as the vector of correlations between the current residual $\mathbf{r}$ and the elements of the dictionary $\mathbf{\Phi} = [\,\varphi_1 \,|\, \cdots \,|\, \varphi_N\,]$, since it contains the elements $g_k = \langle \varphi_k, \mathbf{r} \rangle$.

Algorithm 1 Generic Directional Pursuit

1: **init** $\mathcal{I} := \varnothing,\ \mathbf{r} := \mathbf{y},\ \mathbf{x} := \mathbf{0} \in \mathbb{C}^N$
2: **repeat**
3: $\quad \mathbf{g} := \mathbf{\Phi}^{\mathsf{H}}\,\mathbf{r}$
4: $\quad i := \underset{k}{\operatorname{argmax}}\,|g_k|$
5: $\quad \mathcal{I} := \mathcal{I} \cup \{i\}$
6: $\quad$ **calc update direction** $\mathbf{d}$
7: $\quad \mathbf{c} := \mathbf{\Phi}^{(\mathcal{I})}\,\mathbf{d}_{(\mathcal{I})}$
8: $\quad \alpha := \dfrac{\mathbf{c}^{\mathsf{H}}\,\mathbf{r}}{\|\mathbf{c}\|^2}$
9: $\quad \mathbf{x}_{(\mathcal{I})} := \mathbf{x}_{(\mathcal{I})} + \alpha\,\mathbf{d}_{(\mathcal{I})}$
10: $\quad \mathbf{r} := \mathbf{r} - \alpha\,\mathbf{c}$
11: **until** stop criterion fulfilled
12: **return** $\mathbf{x}$

The index i of the largest absolute value of these correlations is added to the support set $\mathcal{I}$ in line 5. It is worth noting that i can be already in the support set and therefore the support may not necessarily change in each iteration. Also, if it occurs that more than one element has the maximal value, then we choose one of those arbitrarily.
The calculation of the update direction $\mathbf{d}$ in step 6 is the major point where the specific algorithms differ. We will present some examples further on. In step 7 we transform the update direction into 'observation domain'. The restriction to the support set $\mathcal{I}$ in the calculation of $\mathbf{c}$ is not strictly necessary, since $\mathbf{d}$ should be non-zero only on this current support to retain the sparsity during the update. That is, $\mathbf{c} := \mathbf{\Phi}\,\mathbf{d}$ should lead to the same result, but the presented version may avoid unnecessary multiplications with zeros. In line 8 the optimal step size α is calculated, which is then used to perform the update steps for the solution vector $\mathbf{x}$ and the residual $\mathbf{r}$ in lines 9 and 10, respectively. Again, the restriction to the support is not necessary in line 9, but again it may save computations and is stated to emphasize that the current solution is $|\mathcal{I}|$-sparse.
The stopping criterion may be chosen with regard to the specific realization of the algorithm or the problem. Examples may be a maximum number of iterations, some threshold on the residual norm $\|\mathbf{r}\|_2$ or the rate of change of that norm.

As mentioned, there is a long list of greedy algorithms for the sparse reconstruction problem and many of those fall under the class of directional pursuits. We will discuss three important basic algorithms in the following and give an outlook at the end of this clause.

Matching Pursuit

A very basic variant of the directional pursuit is the Matching Pursuit (MP). It was first presented in [65] for time-frequency dictionaries. The calculation of the update direction is simply

$$\mathbf{d} := \mathbf{e}_i \,, \tag{4.1}$$

where i is the index of maximal residual correlation, calculated in step 4 of the generic algorithm. This is a very simple update step, since only that element which belongs to the largest correlation will be changed. With the step size α equal to g_i this yields a very short and efficient implementation, see Algorithm 2.

Computational requirements:

The computational costs per iteration are very moderate in the basic MP algorithm. They are dominated by the calculation of the correlation vector $\mathbf{g} = \mathbf{\Phi}^{\mathrm{H}}\,\mathbf{r}$, which can be done in $\mathcal{O}(MN)$ operations. If for a given sparsity S the requirement $M \in \mathcal{O}(S \log N)$ is fulfilled, we have $\mathcal{O}(S\,N \log N)$ operations. On the other hand, if $\mathbf{\Phi}$ is a sub-matrix of some structured unitary transform where an efficient implementation is available, for instance a DFT sub-matrix, then the complexity reduces to $\mathcal{O}(N \log N)$ operations per iteration, independent of the sparsity S.

However, this simple algorithm requires a very large number of iterations and converges very slowly. This negates the low complexity per iteration.

Algorithm 2 Matching Pursuit

1: **init** $\mathbf{r} := \mathbf{y},\ \mathbf{x} := \mathbf{0} \in \mathbb{C}^N$
2: **repeat**
3: $\quad \mathbf{g} := \mathbf{\Phi}^{\mathrm{H}}\,\mathbf{r}$
4: $\quad i := \underset{k}{\operatorname{argmax}}\,|g_k|$
5: $\quad x_i := x_i + g_i$
6: $\quad \mathbf{r} := \mathbf{r} - g_i\,\boldsymbol{\varphi}_i$
7: **until** stop criterion fulfilled
8: **return** $\mathbf{x}$

Orthogonal Matching Pursuit

The Orthogonal Matching Pursuit (OMP) was originally introduced in [73], see also [78, 79, 70]. It is designed to add a new element to the support in each iteration. This is accomplished by orthogonal projection of the current approximation of $\mathbf{x}$ onto all selected dictionary elements. This on the one hand forces the algorithm to select a new element in each iteration, and on the other hand produces the best approximation achievable with the selected elements in each step, in terms of squared error. The least squares/orthogonalization step is performed in line 5 of Algorithm 3.

The algorithm is written down in a slightly different manner, using the iteration counter k. Due to its properties, the OMP produces a k-sparse approximation in the k-th iteration step. If the sparsity S is known, we can perform exactly S iterations to obtain the best S-sparse solution with the selected elements. If the selection was correct, this leads to the correct solution of the problem.

If the sparsity is unknown one can define another stop criterion, for instance (as mentioned) by a threshold on the residual or on the smallest non-zero element of the actual estimation $\mathbf{x}$. Note that choosing a maximum number of iterations too large may result in overfitting, if the input is noisy or the solution is not exactly sparse. If we are dealing with the noiseless and exactly sparse problem, then the residual is zero after S iterations, provided that we can use exact arithmetic and that the selection of support elements was correct.

Algorithm 3 Orthogonal Matching Pursuit

1: **init** $k := 0,\ \mathcal{I}_0 := \varnothing,\ \mathbf{r}_0 := \mathbf{y},\ \mathbf{x} := \mathbf{0} \in \mathbb{C}^N$

2: **for** $k = 1$ **to** S **do**

3: $\quad i := \underset{j \in \{1,\dots,N\} \setminus \mathcal{I}_{k-1}}{\operatorname{argmax}} \left| \langle \varphi_j, \mathbf{r}_{k-1} \rangle \right|$

4: $\quad \mathcal{I}_k := \mathcal{I}_{k-1} \cup \{i\}$

5: $\quad \mathbf{x}_{(\mathcal{I}_k)} := \underset{\tilde{\mathbf{x}} \in \mathbb{C}^k}{\operatorname{argmin}} \left\| \mathbf{y} - \mathbf{\Phi}^{(\mathcal{I}_k)} \tilde{\mathbf{x}} \right\|_2^2 = \left(\mathbf{\Phi}^{(\mathcal{I}_k)} \right)^+ \mathbf{y}$

6: $\quad \mathbf{r}_k := \mathbf{y} - \mathbf{\Phi}\,\mathbf{x}$

7: **end for**

8: **return** $\mathbf{x}$

Reconstruction performance:

There are several results that state that under some conditions the OMP has similar performance to solving BP. A coherence-based result is the following, which is a direct generalization of Lemma 3.5.

Theorem 4.1 (Tropp, 2004, [78])**.** *Both OMP and BP solve the noiseless sparse reconstruction problem* $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$ *with* $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ *exactly, whenever*

$$M \;<\; \frac{1}{2}\left(1 + \frac{1}{\mu(\mathbf{\Phi})}\right) . \tag{4.2}$$

Further theoretical results are stated for example in [79] and [32] for noiseless reconstruction. In the latter one the following RIP-based result is given.

Theorem 4.2 (Davenport, Wakin, 2010, [32])**.** *Suppose* $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ *satisfies a RIP of order* $S+1$ *with* $\delta_{S+1} < \frac{1}{3\sqrt{S}}$*. Then OMP will reconstruct any* $\mathbf{x} \in \Sigma_S$ *exactly from* $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$ *in* S *iterations.*

Although the results for noiseless reconstruction are similar to those for ℓ_1-minimization, this is not the case for noisy reconstruction. So far it has been shown that $M \gtrsim S^2 \log N$ measurements suffice for noisy reconstruction using the OMP, cf. [32]. In [62] a tighter bound of $M \gtrsim S^{1.6} \log N$ is given, which has still a significant gap to the $S \log N$ bound for ℓ_1-methods. However, empirical evidence suggest that $M \gtrsim S \log N$ also suffice for the OMP in case of noise.

Computational complexity:

In addition to the computational requirements for the MP algorithm we have to solve a least squares step of size S. This might be done in additional $\mathcal{O}(S^2)$ operations per iteration. But as mentioned, the overall number of iterations significantly reduces, compared to MP.

Gradient Pursuit

The Gradient Pursuit (GP), proposed in [8], provides a gradient-based approach for the directional pursuit framework. Suppose we are in some iteration where the current support is $\mathcal{I}$. Where in case of OMP we minimize the quadratic cost function

$$J(\mathbf{x}) \;:=\; \left\|\mathbf{y} - \mathbf{\Phi}^{(\mathcal{I})}\mathbf{x}\right\|_2^2 \tag{4.3}$$

completely in each iteration, we now only want to take a step into the opposite gradient direction. We can rewrite the cost function as

$$\begin{aligned} J(\mathbf{x}) &= \left(\mathbf{y} - \boldsymbol{\Phi}^{(\mathcal{I})}\mathbf{x}\right)^{\mathsf{H}} \left(\mathbf{y} - \boldsymbol{\Phi}^{(\mathcal{I})}\mathbf{x}\right) && (4.4) \\ &= \mathbf{y}^{\mathsf{H}}\mathbf{y} \;-\; \mathbf{y}^{\mathsf{H}}\boldsymbol{\Phi}^{(\mathcal{I})}\mathbf{x} \;-\; \mathbf{x}^{\mathsf{H}}\left(\boldsymbol{\Phi}^{(\mathcal{I})}\right)^{\mathsf{H}}\mathbf{y} \;+\; \mathbf{x}^{\mathsf{H}}\left(\boldsymbol{\Phi}^{(\mathcal{I})}\right)^{\mathsf{H}}\boldsymbol{\Phi}^{(\mathcal{I})}\mathbf{x} && (4.5) \end{aligned}$$

and calculate the gradient with respect to $\mathbf{x}^*$ by means of Wirtinger's calculus[1], which yields

$$\begin{aligned} \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}^*} &= -\left(\boldsymbol{\Phi}^{(\mathcal{I})}\right)^{\mathsf{H}}\mathbf{y} \;+\; \left(\boldsymbol{\Phi}^{(\mathcal{I})}\right)^{\mathsf{H}}\boldsymbol{\Phi}^{(\mathcal{I})}\mathbf{x} && (4.6) \\ &= -\left(\boldsymbol{\Phi}^{(\mathcal{I})}\right)^{\mathsf{H}}\left(\mathbf{y} - \boldsymbol{\Phi}^{(\mathcal{I})}\mathbf{x}\right) . && (4.7) \end{aligned}$$

A careful look at the calculation of the vector $\mathbf{g}$ in step 3 of the generic algorithm shows that this has already been calculated in each iteration. More specifically, the update direction calculates as:

6a: $\mathbf{d} := \mathbf{0}$

6b: $\mathbf{d}_{(\mathcal{I})} := -\dfrac{\partial J(\mathbf{x})}{\partial \mathbf{x}^*} = \mathbf{g}_{(\mathcal{I})}$

This is not explicitly carried out in the implementation where we directly use $\mathbf{g}$ for calculating the update, see Algorithm 4.

Algorithm 4 Gradient Pursuit

1: **init** $\mathcal{I} := \varnothing,\ \mathbf{r} := \mathbf{y},\ \mathbf{x} := \mathbf{0} \in \mathbb{C}^N$
2: **repeat**
3: $\quad \mathbf{g} := \boldsymbol{\Phi}^{\mathsf{H}}\mathbf{r}$
4: $\quad i := \underset{k}{\operatorname{argmax}} |g_k|$
5: $\quad \mathcal{I} := \mathcal{I} \cup \{i\}$
6: $\quad \mathbf{c} := \boldsymbol{\Phi}^{(\mathcal{I})}\mathbf{g}_{(\mathcal{I})}$
7: $\quad \alpha := \dfrac{\mathbf{c}^{\mathsf{H}}\mathbf{r}}{\|\mathbf{c}\|^2}$
8: $\quad \mathbf{x}_{(\mathcal{I})} := \mathbf{x}_{(\mathcal{I})} + \alpha\,\mathbf{g}_{(\mathcal{I})}$
9: $\quad \mathbf{r} := \mathbf{r} - \alpha\,\mathbf{c}$
10: **until** stop criterion fulfilled
11: **return** $\mathbf{x}$

[1] See Appendix C for an introduction, or [57, 85] for further information.

Computational costs and performance:

Compared to the Matching Pursuit, the Gradient Pursuit requires only slightly more computation, namely the calculation and application of the step size α. The large part of the calculation, i.e. that of the gradient $\mathbf{g}$, is also done in MP, where it largely remains unused.

Due to the gradient considerations above, this algorithm has a much better convergence than the MP, with comparable costs per iteration. However, one problem might be that the gradient becomes small after coming sufficiently close to the solution. Therefore it might be an advantage if in this case a final least squares step is performed, to fully minimize the cost function (4.3) on the final (hopefully correct) support set. However, this approach provides a good tradeoff between MP and OMP regarding both number of iterations and per-iteration complexity.

Improvements and Other Directional Pursuit Algorithms

There is a huge research work in improving algorithms of the directional framework. For instance there are many ideas to improve the OMP. One idea is the StOMP (Stagewise OMP, [39]), where in steps 3 and 4 of the OMP not only one element is added to the support, but all above a given threshold. The algorithm runs with a fixed number of iterations, independent of the sparsity S. Obviously, the main question is for the optimal choice of such threshold. Another modification is done in the ROMP (Regularized OMP, [71, 72]), where always S elements are chosen, followed by a regularization step. Other ways to save computational costs are for example solving the orthogonalization only approximately, for instance by iterative procedures not until complete convergence.

A way to improve the OMP by previous knowledge on the support or a better initial solution is presented in [86] and named Mod-OMP (modified OMP) in [44]. After obtaining an initial support set of size $K < S$ from this previous information, the orthogonalization is performed on this support. Thereafter the remaining $S - K$ steps are performed as usual.

For the Gradient Pursuit algorithm, an improvement is presented in [8] in terms of applying the conjugate gradient idea to the directional pursuit framework. The resulting Conjugate Gradient Pursuit (CGP) has much higher computational costs than the GP. Therefore the authors of [8] also presented a computationally feasible approximation, referred to as Approximate Conjugate Gradient Pursuit (ACGP).

There are lots of other ideas, for instance the Subspace Pursuit (SP) [30]. For an overview we refer to [46].

4.2.2 Thresholding Algorithms

Whereas the directional pursuit algorithms create a solution by constructing the support step by step, we will now consider another approach. The idea behind thresholding algorithms is to obtain a solution of desired sparsity (or some given quality measure) in each iteration step by calculating a relaxed (non-sparse) solution and applying a thresholding operator afterwards to achieve sparsity.

Iterative Hard Thresholding

A very simple thresholding algorithm is the Iterative Hard Thresholding (IHT) [9]. In each step it produces an S-sparse estimate of $\mathbf{x}$ using the hard thresholding operator H_S, which is defined as follows.

Definition 4.1. The *hard thresholding operator* H_S, applied to a vector $\mathbf{x}$, sets all but the S largest (in magnitude) elements of $\mathbf{x}$ to zero.

In other words, the hard thresholding operator $\mathrm{H}_S\{\mathbf{x}\}$ produces the best S-sparse approximation $\mathbf{x}_S$ (defined at page 5) to a vector $\mathbf{x}$. With this, the algorithm is essentially a one-line iteration:

Algorithm 5 Iterative Hard Thresholding

1: **init** $\mathbf{x} := \mathbf{0} \in \mathbb{C}^N$ or other more feasible values
2: **repeat**
3: $\quad \mathbf{x} := \mathrm{H}_S\left\{\mathbf{x} + \gamma\, \mathbf{\Phi}^{\mathrm{H}} (\mathbf{y} - \mathbf{\Phi}\mathbf{x})\right\}$
4: **until** stop criterion fulfilled
5: **return** $\mathbf{x}$

The aim of the parameter $\gamma > 0$ is to guarantee stability, the most elementary choice would be $\gamma = 1$. If γ is too large, the algorithm becomes unstable, whereas a too small value causes slow convergence. In [75] a proposal and reasoning is given to replace $\gamma\, \mathbf{\Phi}^{\mathrm{H}}$ by the pseudoinverse $\mathbf{\Phi}^{\mathrm{H}} \left(\mathbf{\Phi}\mathbf{\Phi}^{\mathrm{H}}\right)^{-1}$ in Algorithm 5, referred to as Iterative Hard Thresholding with Inversion. However, this is extremely computationally intense but it can be used to obtain suitable values for γ (e.g. by analysis of the matrix norm) to guarantee stability. Indeed, if $\mathbf{\Phi}$ has very low mutual coherence and unit ℓ_2-norm columns, then $\mathbf{\Phi}\mathbf{\Phi}^{\mathrm{H}}$ is approximately the identity matrix, which would verify $\gamma = 1$ to be the best possible guess. One can also show that the argument of the H_S-operator

in the algorithm is equivalent to the Jacobi iteration[2] for solving the system of normal equations $\mathbf{\Phi}^{\mathrm{H}}\mathbf{y} = \mathbf{\Phi}^{\mathrm{H}}\mathbf{\Phi}\mathbf{x}$, see [50].

The IHT algorithm calculates an S-sparse solution in each step, such that it iteratively refines an existing S-sparse solution, which is also attractive for online estimation. That is, we can use suitable initial solutions, and also we can iterate over several observations $\mathbf{y}$ without any change in the algorithm. Moreover, we can replace the sensing matrix $\mathbf{\Phi}$ (together with the belonging observation vector) from one iteration to the next without any further change, which can be a huge advantage in specific applications.

Complexity and convergence:

To begin with, we take a look at the complexity of the hard thresholding operator $\mathrm{H}_S\{\mathbf{x}\}$, which will also be used in some of the following algorithms. In a simple first approach we can consider sorting the elements of $\mathbf{x} \in \mathbb{C}^N$ in decreasing order, which is possible in $\mathcal{O}(N \log N)$, and then pick the first S elements. There are also $\mathcal{O}(N \log S)$ and $\mathcal{O}(N + S \log S)$ algorithms, for instance by holding a heap of size S. Interestingly, the theoretical complexity of this problem is $\mathcal{O}(N)$, independent of S. The problem can be solved by solving the *selection problem*, see [28] or [58] for detail.

The complexity in each step is further dominated by application of two matrix multiplications with $\mathbf{\Phi}$ and $\mathbf{\Phi}^{\mathrm{H}}$, which is $\mathcal{O}(M N)$ each. For partial structured transform matrices, such as partial DFT matrices, this is possible in $\mathcal{O}(N \log N)$ (theoretically in $\mathcal{O}(N \log S)$, but in conjunction with a higher overhead).

The authors of [9] showed convergence and asymptotic error bounding results similar to those of ℓ_1 techniques. Nevertheless, this algorithm is sensitive to initial solutions and can get stuck in local optima.

Moreover, the convergence speed of IHT is only linear, which is a crucial point. However, there exist accelerated versions of IHT, which will be described in the following.

Accelerated Iterative Hard Thresholding

Due to the slow convergence of the IHT, there exist a couple of improvements to accelerate this algorithm. According to the author of [7], the Accelerated Iterative Hard Thresholding (AIHT) algorithms can be split into two classes.

Variant 1:

When tracing the convergence process of the IHT, one can observe that during the

[2] For an overview on the Jacobi method for linear equations see for instance [53].

Algorithm 6 Accelerated Iterative Hard Thresholding – Variant 1

1: **init** $\mathbf{x} := \mathbf{0} \in \mathbb{C}^N$ or other more feasible values
2: **repeat**
3: $\quad \mathbf{x} := \mathsf{H}_S\left\{\mathbf{x} + \gamma\, \mathbf{\Phi}^{\mathsf{H}}\left(\mathbf{y} - \mathbf{\Phi}\mathbf{x}\right)\right\}$
4: $\quad \mathcal{I} := \mathrm{supp}(\mathbf{x})$
5: $\quad \mathbf{x}_{(\mathcal{I})} := \underset{\tilde{\mathbf{x}} \in \mathbb{C}^{|\mathcal{I}|}}{\mathrm{argmin}} \left\|\mathbf{y} - \mathbf{\Phi}^{(\mathcal{I})}\tilde{\mathbf{x}}\right\|_2^2$
6: **until** stop criterion fulfilled
7: **return** $\mathbf{x}$

iterations the support does not change for a large number of steps, and the solution changes only slightly on this support. This can motivate to look for the least squares solution on the current support in each step, as it is done in the OMP.

This is exactly what is done in the AIHT algorithm of type 1: In each iteration refine the recent estimation $\mathbf{x}$ only on its support. This is done by adding a least squares / orthogonalization step after the hard thresholding step; see Algorithm 6. This was proposed in [48] under the name of Hard Thresholding Pursuit. The subsequent least squares step also occurs in the Iterative Hard Thresholding with Inversion in [64].

Similar to the situation between MP and OMP, the least squares step enlarges the complexity per iteration significantly, whereas on the other hand it dramatically reduces the number of iterations. To mitigate this growth in complexity, one can think of solving the least squares problem iteratively and only accomplish a few such inner iteration steps. This gives a tradeoff between inner (least squares) and outer (IHT) iteration steps.

This variant yields a very simple stop criterion: If the support does not change after thresholding, compared to the previous iteration, then the solution will never change in further iterations. Therefore we can stop the iteration.

Variant 2:

The AIHT of type 1 turns out to have the same problem of local minima as the IHT has. Therefore the type 2 AIHT algorithms allow a relaxation of the thresholding strategy to more than S non-zeros. That is, in each iteration an extension of the support of $\mathbf{x}$ is possible during optimization, followed by a consecutive H_S-thresholding step. One example is the Double Over Relaxation approach (DORE), presented in [75].

These strategies also appear in a larger context of mixtures with other strategies, as we will briefly discuss in the next clause.

4.2.3 Mixtures and Extensions

There are many algorithm proposals that combine ideas from several fields of optimization and heuristics, especially from the ideas of the previous sections. Moreover there are extensions, for instance by solving a sequence of sparse reconstruction problems, related to a single initial problem. This idea is applied for instance in case of the Iterative Support Detection (ISD), described in [83]. This is not a concrete algorithm to solve a sparse reconstruction problem, but rather a framework for iteratively refining a solution. ISD iteratively solves a sequence of truncated sparse reconstruction problems, shrinking a threshold in each iteration.

However, we will restrict to the mixture of Directional Pursuit and thresholding ideas, especially we will consider a very popular algorithm – the CoSaMP.

CoSaMP

The abbreviation CoSaMP stands for Compressed Sampling Matching Pursuit, which implies that this algorithm was specially designed for problems in the field of Compressed Sensing. Its reconstruction guarantees are similar to those of ℓ_1-techniques. It was presented in a previous version of [69] in 2008, see also [70].

The algorithm uses thresholding ideas, dictionary correlation and support merging techniques, as well as the orthogonalization step from the previous subsections. It can be seen as a classical example for an AIHT algorithm of the second variant, since it allows a relaxation to a sparsity of $2S$ during the calculations, see Algorithm 7. It overcomes the disadvantage of OMP that once an index is chosen, it will stay in the support whether it is correct or not. Further it calculates an S-sparse solution in each iteration, due to the hard thresholding step.

The initialization of $\mathbf{x}$ can be also done with a previous estimate, which can improve convergence and is also attractive for online estimation. The relaxation to a sparsity of $2S$ can be generalized to αS for some $\alpha \geq 1$, which may be an advantage for specific sparsity models.

Complexity and convergence:

As for computational effort, the hard thresholding operator is used twice per iteration. In the first use in line 3 it looks for the $2S$ largest elements in a vector of size N, which is possible in $\mathcal{O}(N)$ as stated above. A careful look at the second use in line 6 shows that the search for S largest elements is performed on no more than $3S$ elements, since the

Algorithm 7 CoSaMP

1: **init** $\mathcal{I} := \varnothing,\ \mathbf{r} := \mathbf{y},\ \mathbf{x} := \mathbf{0} \in \mathbb{C}^N$
2: **repeat**
3: $\quad \mathcal{I} := \{\ k,\ \text{belonging to the } 2S \text{ largest values of } |\langle \boldsymbol{\varphi}_k, \mathbf{r} \rangle|\ \} \ \cup\ \operatorname{supp}(\mathbf{x})$
$\qquad = \ \operatorname{supp}\left(\mathsf{H}_{2S}\left\{\boldsymbol{\Phi}^{\mathsf{H}}\,\mathbf{r}\right\}\right) \ \cup\ \operatorname{supp}(\mathbf{x})$
4: $\quad \hat{\mathbf{x}} := \mathbf{0} \in \mathbb{C}^N$
5: $\quad \hat{\mathbf{x}}_{(\mathcal{I})} := \underset{\tilde{\mathbf{x}} \in \mathbb{C}^{|\mathcal{I}|}}{\operatorname{argmin}} \left\|\mathbf{y} - \boldsymbol{\Phi}^{(\mathcal{I})}\tilde{\mathbf{x}}\right\|_2^2 \ = \ \left(\boldsymbol{\Phi}^{(\mathcal{I})}\right)^{+}\mathbf{y}$
6: $\quad \mathbf{x} := \mathsf{H}_S\left\{\hat{\mathbf{x}}\right\}$
7: $\quad \mathbf{r} := \mathbf{y} - \boldsymbol{\Phi}\,\mathbf{x}$
8: **until** stop criterion fulfilled
9: **return** $\mathbf{x}$

support of the input has at most $3S$ elements. This yields an effort of $\mathcal{O}(S)$ for that. The least squares step in line 5 may be performed in $\mathcal{O}(S^2)$ calculations. In summary this yields a complexity per iteration comparable to that of the OMP.

In [69] it is shown that the algorithm yields performance guarantees with a fixed number of iterations, that is $\mathcal{O}(1)$. Depending on the concrete number this can be worse or better than OMP. Another stopping criterion, depending on the residual norm with a precision parameter η, that is, stop if $\|\mathbf{r}\|_2 \leq \eta$, yields a number of $\mathcal{O}\Big(\log \frac{\|\mathbf{x}\|_2}{\eta}\Big)$ iterations.

A huge advantage over OMP is that CoSaMP yields performance guarantees also for noisy measurements and non-exactly sparse input vectors. Comparable to the result for ℓ_1-relaxation in Theorem 3.7, the following holds.

Theorem 4.3 (Needell, Tropp, 2009, [69])**.** *Given a noisy measurement model* $\mathbf{y} = \boldsymbol{\Phi}\,\mathbf{x} + \mathbf{n}$ *as in (2.5) with* $\|\mathbf{n}\|_2 \leq \varepsilon$*. If the measurement matrix* $\boldsymbol{\Phi}$ *obeys a RIP of order* $4S$ *with* $\delta_{4S} < 0.1$*, then after* $\mathcal{O}(S)$ *iterations, CoSaMP produces an estimate* $\hat{\mathbf{x}}$ *satisfying*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \ \leq\ C_0 \frac{\sigma_S(\mathbf{x})_1}{\sqrt{S}} \ +\ C_1\,\varepsilon \tag{4.8}$$

for some positive constants C_0 *and* C_1*.*

4.3 Other Approaches

4.3.1 Non-Convex Relaxation

Minimizing ℓ_1 leads to very efficient methods, but the price we pay for this simplification is a larger required number of measurements, compared to combinatorial search (nevertheless this number is far below the required number for conventional methods). Another approach, for example presented by Rick Chartrand in [22] and [23], uses non-convex relaxation by means of minimizing the ℓ_p-(quasi-)norm with $0 < p < 1$. In this case, compared to ℓ_1-minimization, fewer measurements are required but also the computational complexity increases.

In fact, the results are more general for all $p > 0$. The general optimization problem, as already stated in Section 2.2, reads:

$$(P_{p>0}) \qquad \begin{array}{ll} \text{Minimize} & \|\mathbf{x}\|_p \\ \text{subject to} & \mathbf{y} = \mathbf{\Phi}\,\mathbf{x} \end{array}$$

The result of Theorem 3.6 for the exactly sparse and noiseless case is generalized in [22] in the following way.

Theorem 4.4. *Let $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$, let $\mathbf{x} \in \mathbb{C}^N$ be S-sparse, and let $0 < p \leq 1$, $b > 1$, and $a = b^{\frac{p}{2-p}}$. If $\mathbf{\Phi}$ satisfies the RIP-condition*

$$\delta_{a\,S} + b\,\delta_{(a+1)S} \;<\; b - 1\,, \tag{4.9}$$

a solution $\hat{\mathbf{x}}$ to $(P_{p>0})$ is unique and equal to $\mathbf{x}$.

Fix $b = 3$ and $p = 1$ to obtain the result of Theorem 3.6 for (P_1) with exactly sparse input.

Using this theorem, a generalization of Lemma 3.4, which said that the ℓ_0-problem is uniquely solvable if $\delta_{2S} < 1$, is stated for all $p > 0$ as follows.

Lemma 4.1. *Let $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ and let $\mathbf{x} \in \mathbb{C}^N$ be S-sparse. If $\delta_{2S+1} < 1$, then there is a $p > 0$ such that a solution $\hat{\mathbf{x}}$ to $(P_{p>0})$ is unique and equal to $\mathbf{x}$.*

It should be mentioned that the RIP condition of the theorem holds, if $M > C\,S\log N$ for some positive constant C, and that this lower bound is sharp (cf. [15, 16]). This asymptotic bound is independent of the concrete value of p for ℓ_p-relaxation, but it is suggested that the value of the constant C can be reduced by choosing $p < 1$ compared to ℓ_1-minimization.

The reconstruction method in [22] is based on a simple gradient descent approach with projection, which in case $p < 1$ will only find a local optimum due to non-convexity of the problem. However, the results show that the global optimum can be found by initializing with a point sufficiently close to the solution.

4.3.2 The Smoothed ℓ_0 Algorithm

A completely different approach is presented by Mohimani et al. in [66] and [67]. Herein, the authors introduce an approximation of the discontinuous ℓ_0-'norm'. First, define

$$f_\sigma(x) := e^{-\frac{|x|^2}{2\sigma^2}}, \tag{4.10}$$

for some $\sigma > 0$, and then for $\mathbf{x} = (x_1, \ldots, x_N)^\mathsf{T}$ define

$$F_\sigma(\mathbf{x}) := \sum_{i=1}^{N} f_\sigma(x_i). \tag{4.11}$$

Since

$$\lim_{\sigma \to 0} f_\sigma(x) = \begin{cases} 1, & x = 0 \\ 0, & \text{else} \end{cases} \tag{4.12}$$

it follows

$$\lim_{\sigma \to 0} F_\sigma(\mathbf{x}) = N - \|\mathbf{x}\|_0 . \tag{4.13}$$

This leads to the approximation

$$\|\mathbf{x}\|_0 \approx N - F_\sigma(\mathbf{x}) \tag{4.14}$$

for sufficiently small σ. Approximately solving (P_0) by means of minimizing the approximate ℓ_0-'norm' is equivalent to maximizing $F_\sigma(\mathbf{x})$.

The idea of the algorithm is as follows:

- Begin with a feasible start solution, e.g. the ℓ_2-solution.
- Define a sequence of decreasing values of σ, from sufficiently large to sufficiently small.
- For each value of σ improve the current solution by using a gradient ascent algorithm to maximize $F_\sigma(\mathbf{x})$.

The final solution for small σ is an approximation to the ℓ_0-solution.

If decreasing σ sufficiently slow, the hope is that we avoid getting stuck in a local

optimum. We can observe a relation to machine learning techniques in this approach, since we are somehow 'learning' the ℓ_0-'norm'.

The authors of [66] and [67] also showed that for $\sigma \to \infty$, the maximization of $F_\sigma(\mathbf{x})$ is equivalent to minimizing the ℓ_2-norm of the original problem. This justifies the start with the ℓ_2-solution as '$\sigma = \infty$ - solution' and then decreasing σ to arbitrarily close to zero. Moreover, the authors showed that their algorithm is also robust to noise, which the original ℓ_0-problem is not. The modification is to decrease σ only down to the noise floor, in order to not obtain non-zeros caused by the noise.

5 Numerical Examples

5.1 Noiseless Reconstruction

In this chapter a small selection of experiments is presented to illustrate the performance of several reconstruction algorithms. We begin with the noiseless case and two selected examples. The notation is the standard one of this thesis, i.e. we consider the measurement equation $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$, with $\mathbf{\Phi} \in \mathbb{C}^{M\times N}$ and $\mathbf{x}$ is S-sparse.

5.1.1 Selected Examples

Gaussian Measurements

In the first example we choose $N = 10000$, $S = 10$ and $M = 100$. The S non-zero elements in $\mathbf{x}$ are i.i.d. standard Gaussian distributed and are placed at random positions. The entries of $\mathbf{\Phi}$ are i.i.d. realizations of a zero-mean Gaussian distribution with variance $\frac{1}{M}$.

Reconstruction results:

Figure 5.1 shows the original vector $\mathbf{x}$ and its reconstruction $\hat{\mathbf{x}}$, obtained by ℓ_1-minimization. There is no visible difference between both, the normalized squared error, defined as

$$\text{NSE} \;:=\; \frac{\|\mathbf{x}-\hat{\mathbf{x}}\|_2}{\|\mathbf{x}\|_2} \tag{5.1}$$

has a value of NSE $\approx 10^{-10}$, which corresponds with the stopping threshold of the solver. The applied ℓ_1-solver is the ℓ_1-MAGIC by Candès and Romberg [11].

The same problem instance was also solved with OMP, GP and CoSaMP. The error was always below NSE $= 10^{-9}$ and all of these greedy algorithms detected the support correctly. The OMP was carried out with exactly $S = 10$ iterations, whereas CoSaMP reached the same precision within only 5 iterations. On the other hand, the GP algorithm needed 35 iterations for convergence, but recall that its computational costs per iteration are much smaller.

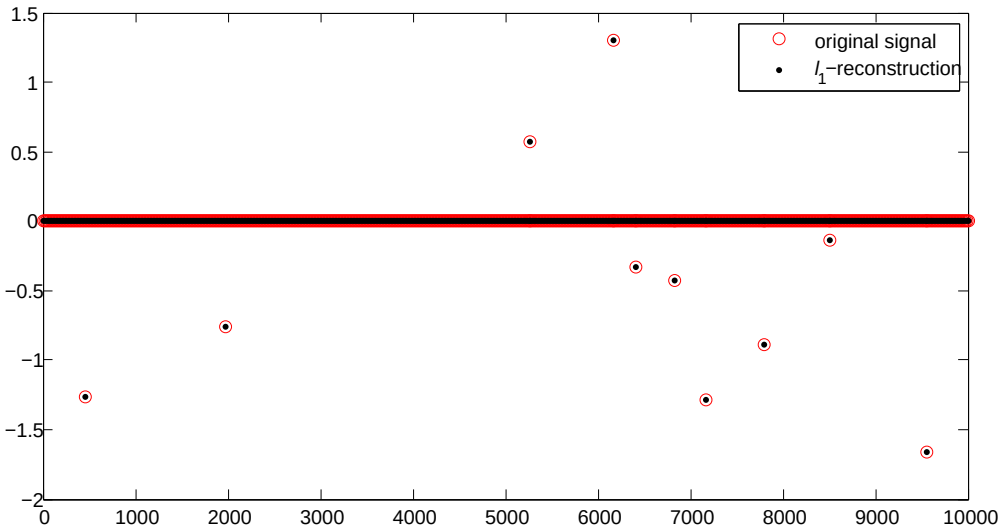


Figure 5.1: Noiseless reconstruction of a vector of length 10^4 with 10 non-zero elements from 100 Gaussian measurements. That is 1 % measurements compared to the signal dimension.

The computation time of the greedy algorithms, compared to that of the ℓ_1-solver, are:

- OMP: 11%
- GP: 7%
- CoSaMP: 3%

Of course, these values are of limited validity due to implementation issues, but they give a hint. At least observe that the greedy algorithms are significantly faster.

It should be mentioned that the IHT of variant 1 did not lead to success in this example, except for initialization with random values on the correct support.

Results for the smoothed ℓ_0 approach:

The smoothed ℓ_0 algorithm (SL0) from Section 4.3.2 highly depends on the parameters, given to the algorithm. Especially, it depends on the decay rate of the parameter σ, which decreases by the law

$$\sigma^{(k)} := r\,\sigma^{(k-1)} \tag{5.2}$$

in the k-th iteration step, where $0 < r < 1$ is the decrease factor. The decay is continued until a final value σ_{final} is reached, which determines the precision of the algorithm. This value was set to 10^{-6} in this simulation.

The default value for the decay rate is set to $r = \frac{1}{2}$ in the package provided by the authors of [67]. The higher r is, the better is the result, but also the larger is the computation time. The error values for different r, together with the relative computation time

σ-decrease factor r	normalized squared error	relative commutation time $t(\text{SL0})/t(\ell_1)$
0.9	0.44	1.2
0.95	0.025	2.0
0.96	7×10^{-4}	2.5
0.97	5.5×10^{-6}	3.3
0.98	1.3×10^{-6}	4.9
0.99	7.8×10^{-7}	9.7

Table 5.1: Performance of the smoothed ℓ_0 algorithm.

(compared to ℓ_1-minimization) is stated in Table 5.1. Useful reconstruction results are achieved not until a rate r significantly above 0.9, which results in a computation time above that of the ℓ_1-solver. Also, the value of r is hard to predetermine, which makes it hard to use this method efficiently. Nevertheless, it is an interesting approach.

Frequency Measurements

We now consider the case of incomplete frequency information as an example of the non-linear sampling theorem stated in Section 3.1.4. This has several practical applications, for instance in magnetic resonance imaging (MRI), channel estimation in mobile radio communication, or remote sensing.

Let $\mathbf{F}$ be the normalized $N\times N$-DFT matrix with entries $(\mathbf{F})_{k,l} = \frac{1}{\sqrt{N}} \exp\left(-2\pi \mathrm{i}\, \frac{k\,l}{N}\right)$, and $\mathbf{F}^{-1} = \mathbf{F}^{\mathrm{H}}$ the inverse transform matrix with entries $\left(\mathbf{F}^{\mathrm{H}}\right)_{k,l} = \frac{1}{\sqrt{N}} \exp\left(2\pi \mathrm{i}\, \frac{k\,l}{N}\right)$, $k, l = 0, \ldots, N-1$. Suppose $\mathbf{x}$ is an S-sparse vector which we interpret as time domain representation, and $\mathbf{F}\,\mathbf{x}$ is its frequency domain interpretation. Given all frequency components, one can reconstruct $\mathbf{x}$ by simply applying the inverse transform $\mathbf{F}^{\mathrm{H}}$. Now suppose that we are only given a small fraction, say M, of these frequency components and want to reconstruct $\mathbf{x}$ using results from Compressed Sensing theory. Or more precisely, we want to recover the complete frequency information from this small fraction, under the assumption that the related time domain representation is sparse. This is often the case in real-world applications.

The parameters in this specific example are $N = 1024$, $M = 50$ and $S = 10$. Denote $\mathbf{u} = \mathbf{F}\,\mathbf{x}$ the unknown vector of complete frequency information, and $\mathcal{M} \subset \{\,0, \cdots, N-1\,\}$ with $|\mathcal{M}| = M$ the subset of indices where frequency information in available. That is, $\mathbf{y} = \mathbf{u}_{(\mathcal{M})}$ is the observation vector and $\mathbf{\Phi} = \mathbf{F}_{(\mathcal{M})}$ is the measurement matrix, and the task is to get an estimate $\hat{\mathbf{u}} = \mathbf{F}\,\hat{\mathbf{x}}$, where $\hat{\mathbf{x}}$ is the solution of the convex problem (P_1).

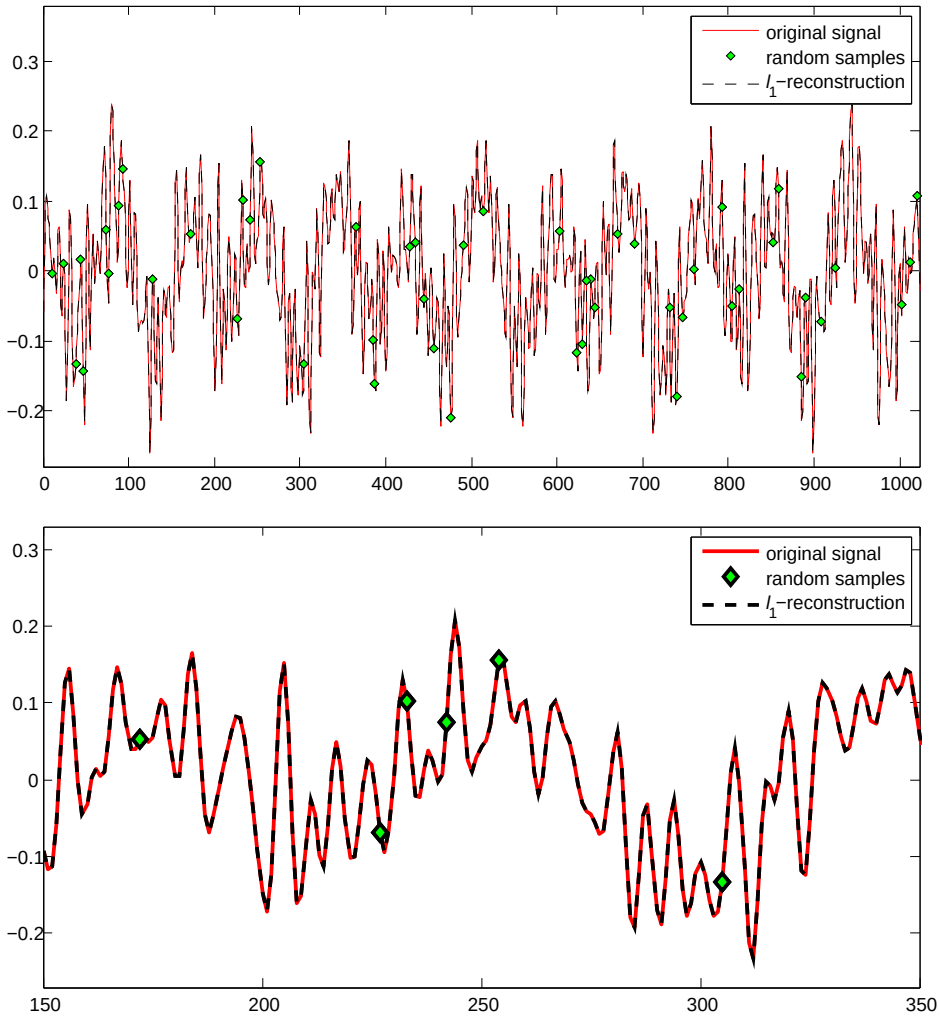


Figure 5.2: Perfect reconstruction of the frequency information from less than 5 % random samples, the plot below shows a zoomed range.

Figure 5.2 shows the real part of the original frequency domain signal, together with the random samples from $\mathcal{M}$ and the reconstruction obtained from that. Again observe perfect match between original and reconstructed signal.

This whole example also works the other way when changing the roles of time and frequency and using the subsampled IDFT matrix for sensing, i.e. $\mathbf{\Phi} = \left(\mathbf{F}^{\mathrm{H}}\right)_{(\mathcal{M})}$. The interpretation of that is the reconstruction of a periodic time domain signal under the assumption that its spectrum is sparse.

Moreover, this is an example of measurements in an incoherent basis and it also works if $\mathbf{x}$ is sparse in an arbitrary basis and the measurements are incoherent and sufficiently many. This is exactly what Section 3.1.4 tells us.

5.1.2 Failure of Least Squares Reconstruction

To illustrate the unsuitability of the least squares approach for sparse reconstruction we consider the following setup. The sensing matrix $\mathbf{\Phi}$ consists of $M = 200$ rows and $N = 1000$ columns with i.i.d. Gaussian entries, and the unknown vector $\mathbf{x}$ has $S = 40$ non-zero entries at random positions, drawn from a standard Gaussian distribution.

We compare the following two solutions. At first, we solve the sparse reconstruction problem via ℓ_1-minimization, and secondly we solve the under-determined system $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$ in the least squares sense using the Moore-Penrose pseudoinverse, i.e. we calculate $\hat{\mathbf{x}} = \mathbf{\Phi}^{+}\mathbf{y}$.

Both solutions are depicted in Figure 5.3, where again the ℓ_1-solution provides exact reconstruction. The least squares solution on the other hand is far from being sparse, the minimum energy approach leads to lots of small entities, but no zeros. Recall the geometric reason in Section 2.2.3 and that least squares is equivalent to ℓ_2-minimization.

In this experiment the ratio $\frac{M}{N}$ was chosen relatively large to make the least squares solution visible. Since the energy of the least squares solution distributes among all N entries, each one has a very small value if $M \ll N$. This effect is already visible in Figure 5.3.

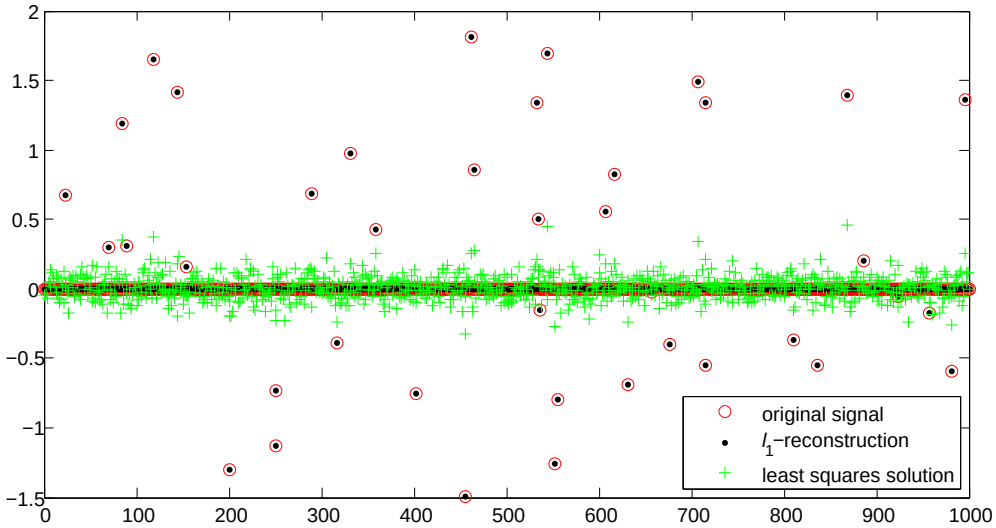


Figure 5.3: ℓ_1 vs. least squares (ℓ_2) reconstruction.

5.1.3 Donoho-Tanner Phase Transitions

An interesting phenomenon is the appearance of a phase transition when varying the sparsity level and the aspect ratio of the measurement matrix. The observation was made by Donoho and Tanner in [38] and connected with other statistical problems and high-dimensional combinatorial geometry.

Figure 5.4 shows such a phase transition diagram, obtained by randomly generating sparse reconstruction problem instances of size $M \times N$, $N = 200$, with Gaussian matrices and solving them via ℓ_1-minimization. The horizontal axis represents the aspect ratio $\delta = \frac{M}{N}$ of the matrix, and the vertical axis shows the ratio of sparsity compared to the number of measurements $\varrho = \frac{S}{M}$. One can clearly observe the phase transition from the lower right part, where ℓ_1-minimization is successful, to the upper left part, where reconstruction is impossible. The black part of the diagram is thresholded by a maximal NSE value of 0.1 (i.e. 10 % error) to make the transition more distinct

There are many open questions concerning the universality of these phase transitions, since all these findings are based on empirical experiments. However, the authors of [38] actually found a function, derived from combinatorial geometry, that perfectly matches the area of phase transition.

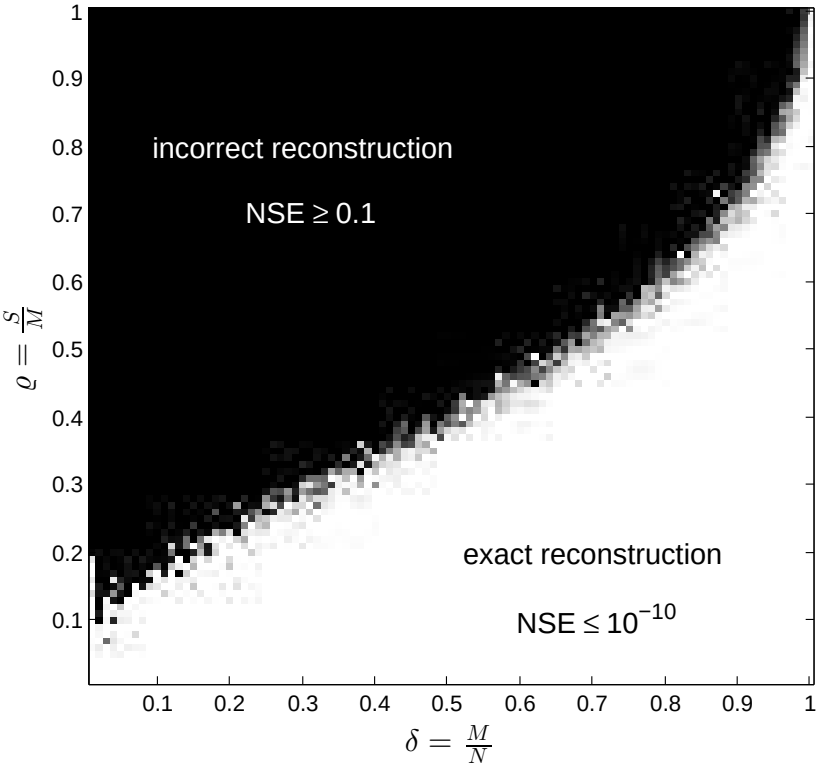


Figure 5.4: Phase transition diagram for ℓ_1-reconstruction.

5.2 Noisy Measurements

In the case of noisy measurements we stick with the measurement equation

$$\mathbf{y} \;=\; \mathbf{\Phi}\,\mathbf{x} + \mathbf{n} \tag{5.3}$$

where $\mathbf{n}$ is an M-dimensional vector with i.i.d. zero-mean Gaussian entries with variance $\sigma_{\mathbf{n}}^2$. We restrict to $M \times N$ Gaussian measurement matrices and S-sparse input vectors with Gaussian entries at random positions.

To make the results comparable we define a criterion for the quality of the measurements. Since position and value of the non-zero elements of $\mathbf{x}$ are arbitrary and $\mathbf{\Phi}$ has a random structure, the entries of the vector $\mathbf{\Phi}\,\mathbf{x}$ can be seen as random variables. These entries are assumed to be zero-mean (by construction of the herein presented examples) and their variance is denoted as $\sigma_{\mathbf{\Phi x}}^2$. The *signal-to-noise ratio* (SNR) of the observation vector $\mathbf{y}$ is defined as

$$\mathrm{SNR} \;:=\; \frac{\sigma_{\mathbf{\Phi x}}^2}{\sigma_{\mathbf{n}}^2}\,. \tag{5.4}$$

For fixed statistics of $\mathbf{\Phi}\,\mathbf{x}$ in the following examples, and varying only the noise level, the SNR is essentially proportional to $\frac{1}{\sigma_{\mathbf{n}}^2}$. We will also use the logarithmic scale, i.e. if the SNR value is stated in dB we mean $10\log_{10}\mathrm{SNR}$.

For sparse estimation via ℓ_1-minimization we have to define a noise bound ε for $\|\mathbf{n}\|_2$. Since $\mathbf{n}$ is random, we set $\varepsilon := \sqrt{M\,\sigma_{\mathbf{n}}^2}$, which is an upper bound on the expected norm of $\mathbf{n}$.

5.2.1 An example for Illustration

To illustrate the reconstruction in presence of noise we choose $N = 1000$, $M = 200$, $S = 10$ and $\mathrm{SNR} = 10\,\mathrm{dB}$. The reconstruction with the ℓ_1-solver and the OMP is depicted in Figure 5.5. The OMP was carried out with exactly S iterations and the ℓ_1-solver used was the CVX package [29, 54] for convex optimization problems.

It can be observed that the ℓ_1-solver finds all non-zero elements, but also finds some additional non-zeros, caused by the noise. Therefore it loses some energy on the correct non-zeros. The observed NSE has a value of 0.11.

The OMP identified all non-zeros correctly. Due to the additional knowledge on the sparsity S the OMP solution contains no additional non-zeros, therefore the solution on this correct support is better than that of the ℓ_1-solution. The observed NSE in this

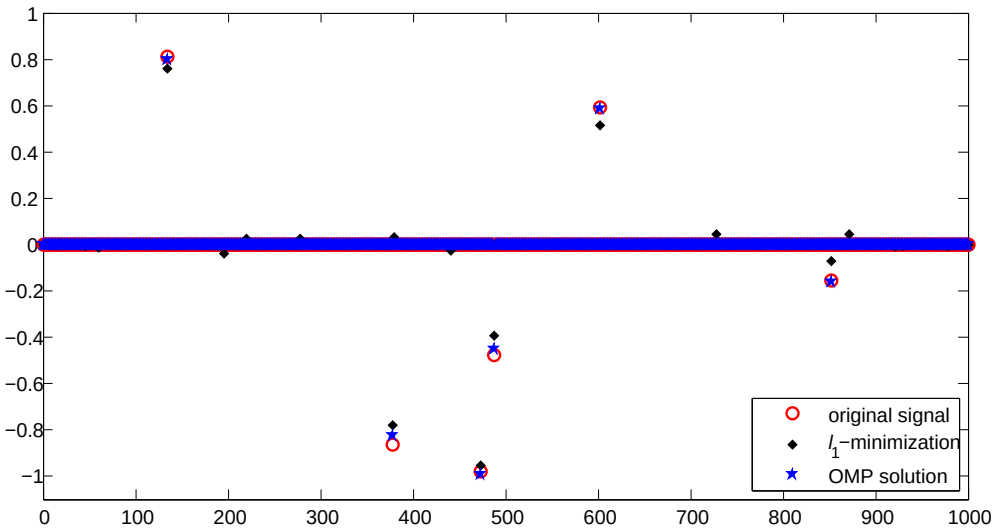


Figure 5.5: Reconstruction from noisy measurements with SNR = 10 dB using ℓ_1-minimization and the OMP with S iterations.

case has a value of 0.03.

5.2.2 Varying Noise Level

For comparison of the results in dependence of the SNR the parameters are set to $N = 1000$, $M = 200$ and $S = 10$. The SNR is varied in a range from 0 dB to 30 dB. The results are obtained by 100 times creating a problem instance randomly with the above parameters, adding noise of different levels, and solving it with each presented algorithm.

The solvers used are the CVX package for the ℓ_1-problem, own implementations of OMP, CoSaMP and IHT/AIHT variant 1, and an implementation of an AIHT of variant 2 from the 'sparsify' software package by the author of [7], provided in [6] (specifically the routine `AIHT.m`). All greedy solvers had knowledge on the sparsity S.

Figure 5.6 shows the NSE in dependence of the SNR on the logarithmic scale for the different solvers. First, one can observe a linearly decreasing behavior on the logarithmic scale when the noise decreases. This verifies that these sparse estimation methods are robust to noisy observations.

When comparing the results for the different solvers, we see that the ℓ_1-solver performs worst, but note that this solver has no knowledge on the sparsity. The simple IHT suffers from outliers due to local minima, which increases the average NSE and makes it the worst performing greedy algorithm here. Although the CoSaMP is an optimal

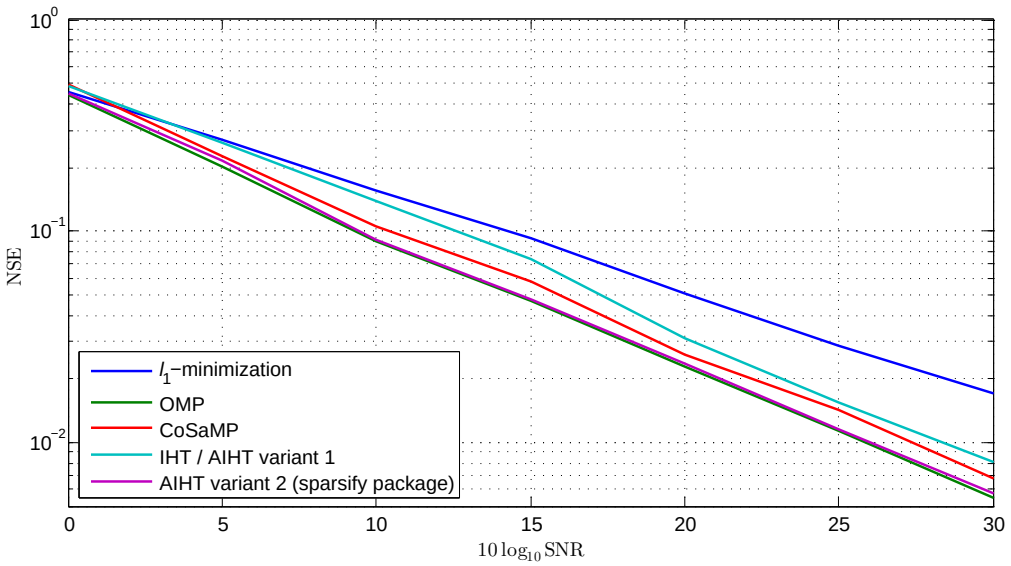


Figure 5.6: Normalized squared error of the solutions for different solvers in dependence of the SNR in dB.

algorithm, it performs worse than the simpler OMP, which led to the best results in these simulations. Very close to the OMP are the results for the AIHT from the sparsify package, which uses some additional heuristics.

5.2.3 Varying Number of Measurements

We want to study the effect of the number of measurements M to the quality of the estimation. The other parameters are again set to $N = 1000$ and $S = 10$, and the SNR varies from 0 dB to 30 dB. The results are again averaged over 100 random problem instances each.

Figure 5.7 shows the reconstruction results for ℓ_1-minimization and for the OMP. As expected, the NSE decreases with increasing M. Also observe that for $M = 25$ there is a very high error floor, which is due to the fact that the requirements for reconstruction from Compressed Sensing theory are not met. That means, no reconstruction is possible with $M = 25$ measurements in this case, and it also means that the phase transition from Section 5.1.3 (generalized for noisy reconstruction) occurs somewhere between 25 and 50 measurements in this example.

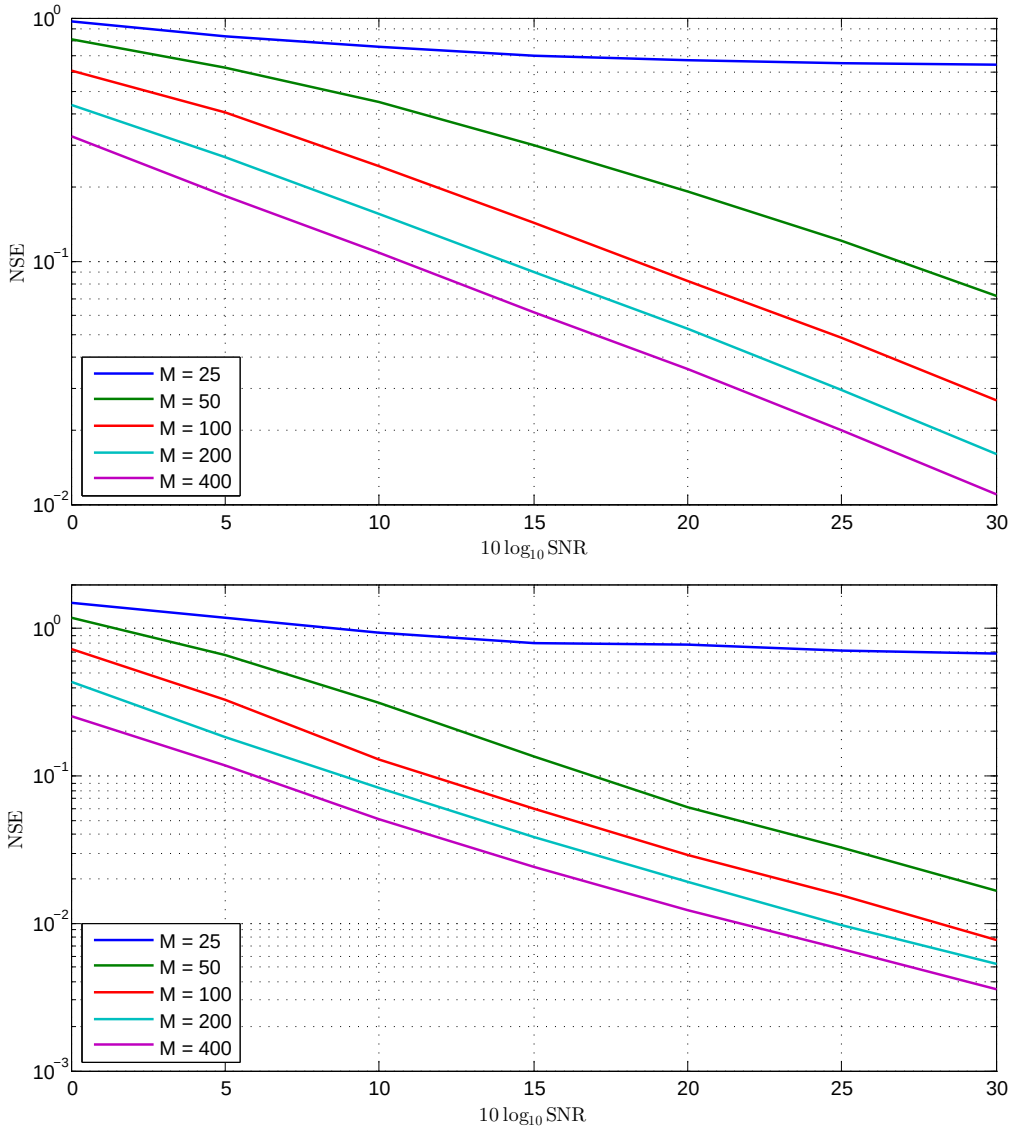


Figure 5.7: Noisy reconstruction results for different numbers of measurements for ℓ_1-minimization (top) and for the OMP (bottom).

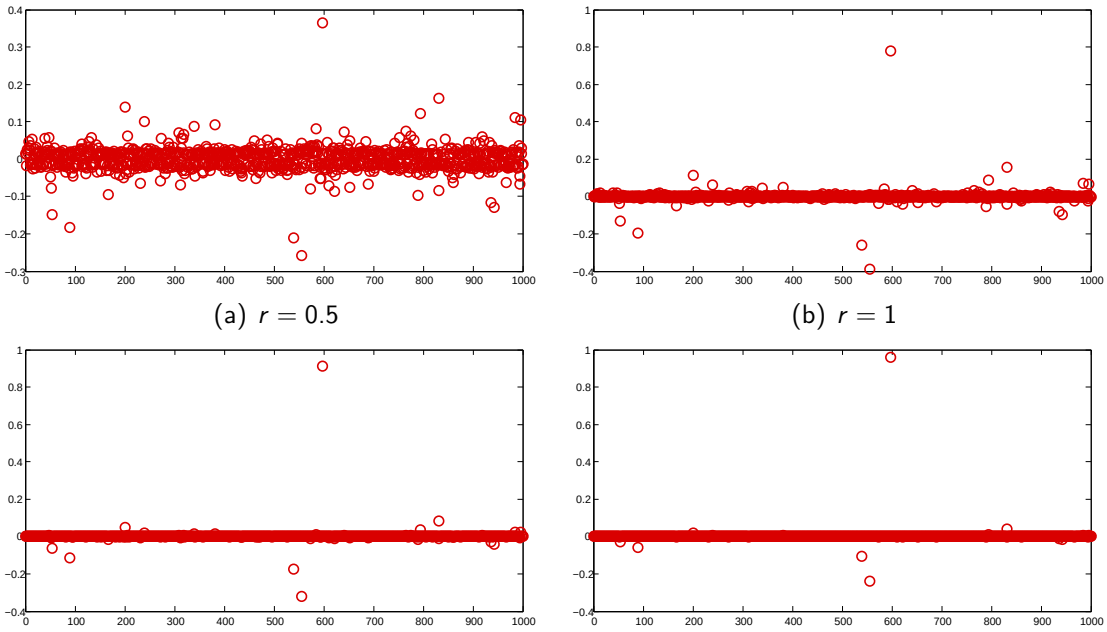
5.3 Approximation of Compressible Signals

We want to briefly illustrate the reconstruction of compressible signals, where we restrict to vectors that obey a power law decay of some rate r, cf. Definition 3.1. From Lemma 3.1 we know that these vectors are 2-compressible with rate $r + \frac{1}{2}$.

We generate reference vectors $\mathbf{x}$ of length N that satisfy the power law decay of different rates r with equality, that is

$$|x_{i_k}| \;=\; C\,k^{-r}\,, \tag{5.5}$$

where the i_k are a random permutation of the sequence $(1,\ldots,N)$ and the sign of the x_{i_k} is chosen independently at random with equal probability. The constant C results



(a) $r = 0.5$ (b) $r = 1$

(c) $r = 1.5$ (d) $r = 2$

Figure 5.8: Power law decay signals with different rates r.

from normalization to unit ℓ_2-norm and reads

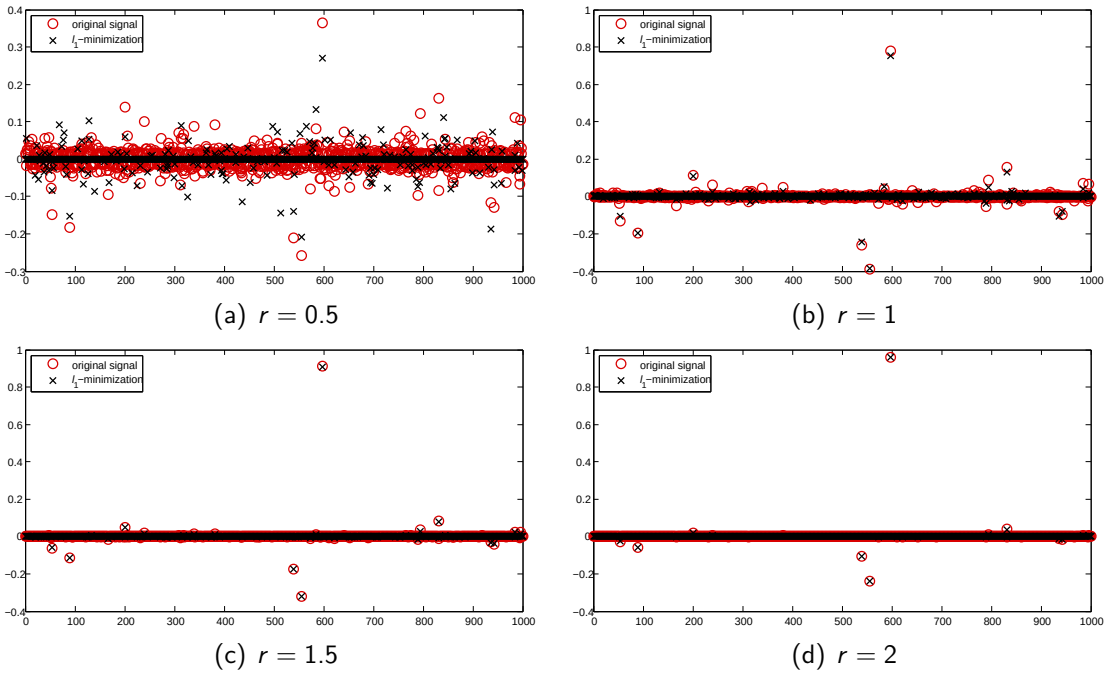
$$C = \left(\sum_{k=1}^{N} k^{-2r} \right)^{-2}.$$

Figure 5.8 illustrates the so constructed $\mathbf{x}$ of length $N = 1000$ for different values of the rate r. To make the plots visually comparable we fixed the sign sequence and permutation and varied only the rate. From a visual point of view one can observe that for a rate $r > 1$ the signals appear very sparse, whereas the signal with $r = 0.5$ does not look compressible at all.

5.3.1 ℓ_1 - Reconstruction

To illustrate the results of ℓ_1-reconstruction we use the vectors $\mathbf{x}$ of length N as constructed above, and we set $M = 200$ and calculate $\mathbf{y} = \mathbf{\Phi}\,\mathbf{x}$ with an $M \times N$ Gaussian sensing matrix $\mathbf{\Phi}$. The reconstructions are obtained by solving the noiseless problem (P_1) with equality constraints.

For visualization, the results are depicted in Figure 5.9 for different rates r. Visually, one can observe that for the signals with $r > 1$ the reconstructions approximate the true signal quite well. As expected, the solution for the signal with $r = 0.5$ is very poor,



(a) $r = 0.5$ (b) $r = 1$

(c) $r = 1.5$ (d) $r = 2$

Figure 5.9: Illustration of ℓ_1 reconstruction of power law decay signals with different rates r.

the NSE is comparable to that of the least squares solution. In this specific example, both had an NSE of ≈ 0.9. The signal with $r = 1$ seems to be a marginal case.

An interesting observation is that the reconstructed signals are M-sparse (to working precision), which might be due to the fact that the ℓ_1-solver solves a problem with M equality constraints.

To compare the quality of the reconstructions we calculate the NSE of the solutions for the above values of r and for different numbers of measurements M. The results are listed in Table 5.2, where the values were obtained by averaging over 100 realizations of the random matrix $\mathbf{\Phi}$. The table also contains the error values of the ℓ_2-solutions (i.e. least squares).

At first, one can observe that the NSE of the least squares solution decreases only slightly with increasing number of measurements, and it is not affected by an increasing rate r. The ℓ_1-solutions on the other hand get significantly better with increasing both M and r.

As already stated above, in case $r = 0.5$ the results of the ℓ_1-minimization are very poor. Their NSE of the is in the same order of magnitude as that of the least squares solution, which extends here to a wider range of M.

	$M = 50$		$M = 100$		$M = 200$		$M = 400$	
	ℓ_1	ℓ_2	ℓ_1	ℓ_2	ℓ_1	ℓ_2	ℓ_1	ℓ_2
$r = 0.5$	1.069	0.973	1.009	0.948	0.904	0.895	0.746	0.778
$r = 1$	0.536	0.975	0.352	0.950	0.216	0.892	0.118	0.776
$r = 1.5$	0.188	0.974	0.082	0.950	0.033	0.897	0.011	0.773
$r = 2$	0.072	0.975	0.018	0.948	$4.6{\cdot}10^{-3}$	0.894	$9.4{\cdot}10^{-4}$	0.778

Table 5.2: NSE of power law decay signal reconstruction, r vs. M.

5.3.2 Reconstruction Using a Greedy Algorithm

To study the reconstruction performance of a greedy method we will restrict to the OMP algorithm (Alg. 3, page 39), which already proved good performance for exactly sparse inputs with and without noise.

However, since we cannot specify the sparsity S for a compressible input, we have to modify the stopping criterion. We do this by bounding the residual norm by some threshold t. That is, we replace the FOR loop in Algorithm 3 by a REPEAT loop and iterate until

$$\|\mathbf{r}\|_2 \;\leq\; t\,\|\mathbf{y}\|_2\,. \tag{5.6}$$

The threshold $0 < t < 1$ has to be close to zero and approximately bounds the allowed approximation error. This can be motivated with the matrix $\mathbf{\Phi}$ satisfying some RIP, and therefore, if $\mathbf{x}$ can be well approximated by some sparse vector, it acts nearly like an isometric map on that $\mathbf{x}$. Therefore, the bound

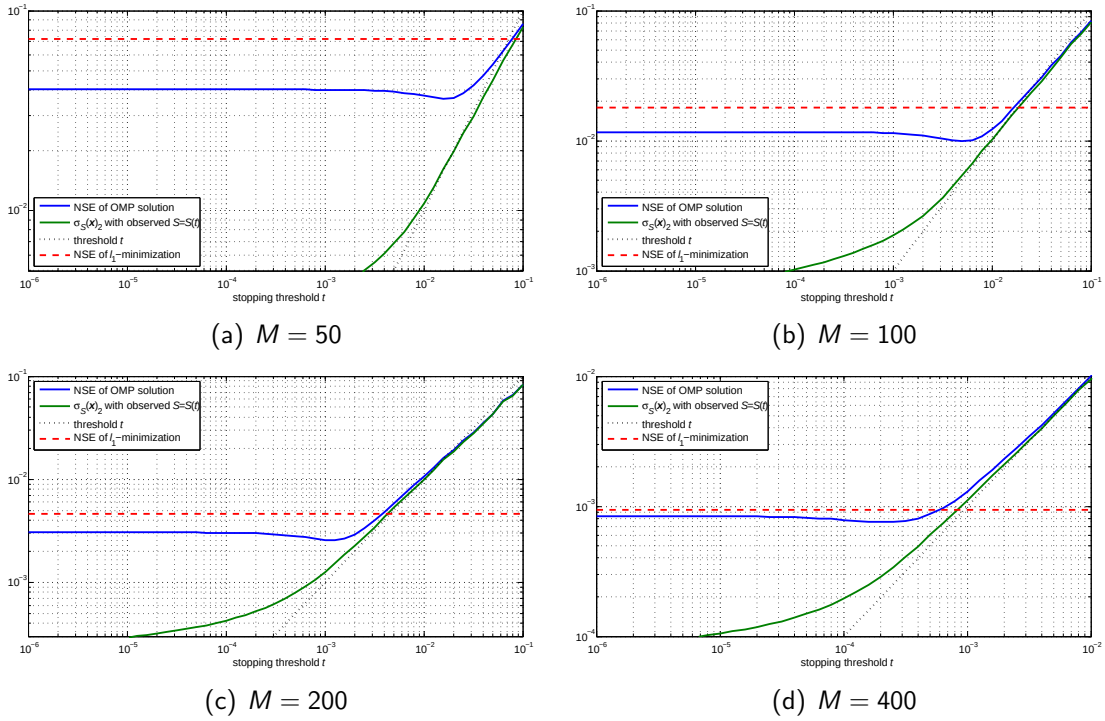
$$t\,\|\mathbf{y}\|_2 \;\geq\; \|\mathbf{r}\|_2 \;=\; \|\mathbf{y} - \mathbf{\Phi}\,\hat{\mathbf{x}}\|_2 \;=\; \|\mathbf{\Phi}\,\mathbf{x} - \mathbf{\Phi}\,\hat{\mathbf{x}}\|_2 \;\approx\; \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \tag{5.7}$$

together with $\|\mathbf{y}\|_2 \approx \|\mathbf{x}\|_2$ results in some approximate error

$$\frac{\|\mathbf{x} - \hat{\mathbf{x}}\|_2}{\|\mathbf{x}\|_2} \;\lesssim\; t\,. \tag{5.8}$$

Also, if the input is noisy, the threshold may be chosen with regard to the noise floor, in order to avoid non-zeros caused by the noise.

At first we want to study the effects of the parameter t to the results in terms of error. For this purpose we use the size $N = 1000$ as before and set $r = 2$ and vary the threshold t in the range $[10^{-6}, \ldots, 10^{-1}]$. Figure 5.10 illustrates the results for different numbers of measurements M, where again the results are averaged over 100 realizations of the matrix $\mathbf{\Phi}$. The sparsity of a solution depends on the parameter t, and with the resulting



(a) $M = 50$ (b) $M = 100$

(c) $M = 200$ (d) $M = 400$

Figure 5.10: Error of OMP reconstruction, depending on the stopping threshold t for different numbers of measurements M.

sparsity $S = S(t)$, the error $\sigma_S(\mathbf{x})_2$ of the best possible S-sparse approximation is also plotted. Additionally the NSE values of the ℓ_1-results from Table 5.2 are plotted. The first thing that attracts attention is that for small $t \to 0$ the NSE of the OMP solution runs into an error floor, whereas for values $t \to 1$ we have NSE $\sim t$ (i.e. Inequality (5.8) holds approximately with equality). In the transition area between those two cases there is a minimum NSE value and the respective t would be the optimal choice. This might be hard to determine in practical cases, it appears in the decade $(0.1, \ldots, 1) \times$ the error floor, translated to the t-axis. The error floor itself depends on the number M of measurements. Note that for this choice of r the OMP always performs better than ℓ_1-minimization, if the threshold is small enough.

To study the effects of a varying decay rate r we fix the stopping threshold of the OMP to $t = 10^{-3}$ and set $M = 200$. The rate r varies in a range from 0.5 to 2 like in the previous section. Table 5.3 shows the NSE of the OMP solutions compared to the values of the ℓ_1 solutions. As already observed in the previous section, for rates $r < 1$ the results are very poor, and for $r = 1$ there is some small gain compared to least squares (which has NSE ≈ 0.9, cf. Table 5.2) but the result is not really usable. Here ℓ_1 performs slightly better, but for $r \geq 1.5$ the OMP outperforms ℓ_1-minimization.

Table 5.3 also shows the observed sparsity of the OMP solutions, which depends on the stopping parameter t (here fixed) and the rate r. Depending on these two parameters

	NSE OMP	NSE ℓ_1	observed sparsity S	best S-term approximation error $\sigma_S(\mathbf{x})_2$
$r = 0.5$	1.2528	0.9038	160	0.4946
$r = 1$	0.2637	0.2155	146	0.05952
$r = 1.5$	0.03055	0.03307	116	0.005516
$r = 2$	0.002638	0.004596	60	0.001225

Table 5.3: Comparison of the NSE of OMP vs. ℓ_1-minimization for different r.

the OMP produces a much sparser solution than ℓ_1-minimization where we observed $S = M$. The rightmost column of the table shows the hypothetical error that could be reached with the same number of non-zero elements as the OMP result has, obtained by simply thresholding the original input $\mathbf{x}$. Note that even if the OMP identifies the same support (i.e. the indices of the largest elements, which it actually did find in the experiments), it returns the least squares solution on this support, which is different from the original input. This is due to the effect that the remaining elements have on the observation vector. However, this deviation decreases the higher r is, which can be seen in the last row of the table.

One can also define other stopping thresholds, for instance if the smallest non-zero element of the current solution falls below a predefined threshold. Simulations showed that this strategy leads to similar results as those depicted in Figure 5.10 under the same conditions. The behavior was the same and the error floor for small thresholds was similar, therefore we omit discussing details for that strategy.

The determination of good stopping thresholds is difficult in practice since it varies with the unknown vector $\mathbf{x}$ and the number of measurements. For power law decay signals, if the rate of decay r is known, it can be used to define the threshold t, but usually it is unknown. A general rule is that the threshold should be chosen rather too small than too large. Note that this does not hold if the input is additionally noisy. Then the threshold has to be chosen with regard to the noise level, since if chosen too small it may result in significant non-zero entries caused by noise.

From the observations in this section we can conclude that a power law decay vector can be called compressible if $r > 1$. In this case sparse reconstruction methods lead to usable results with much smaller error than least squares reconstruction.

6 Summary and Outlook

In the previous chapters we have given an overview to the novel and vastly growing theory of Compressed Sensing. Starting with the $\mathcal{NP}$-hard sparse reconstruction problem in Chapter 2 we discussed the formulation as ℓ_0-minimization problem, interpreting the objective of sparsity as an extreme case of the ℓ_p-norm. This led straightforwardly to a possibility of optimization-theoretical relaxation, namely in the ℓ_p sense. In particular we considered the ℓ_1-relaxation as convex relaxation, and we observed that the ℓ_2-relaxation matched the conventional least squares problem. A geometrical look at these two problems gave reason that a solution to the ℓ_1-problem might be sparse, whereas the least squares approach turned out to be an inappropriate model for finding sparse solutions. The same observations were made for the modified sparse reconstruction problem with an additional noise term.

A more theoretical foundation of the ℓ_1-approach was presented in Chapter 3, where some interesting insights were presented regarding the requirements for the equality of solutions to the ℓ_0 and the ℓ_1 problem. This is related to the sensing process, which is the origin of the problem. We stated requirements for the sensing process, for instance that measurements have to be incoherent to the sparsity basis of the signal, and how many measurements are necessary. With several quality measures for the sensing matrix, especially the RIP, it turned out that the best case is a purely random matrix, yielding the best results in terms of reconstruction error and minimum number of measurements – with high probability. There are many open questions in the area of sensing matrix construction, since there is still a huge gap in performance guarantees between deterministically constructed and random sensing matrices.

From another viewpoint we have observed in Chapter 4 that there exist a lot of other methods to approach the sparse reconstruction problem. Apart from the whole ℓ_1 considerations, these methods try to solve the problem directly. Especially for the example of iterative algorithms using greedy strategies we have seen that there are similar performance guarantees as available for ℓ_1 methods. This was also verified by the simulation results in Chapter 5.

Although the general sparse reconstruction problem is $\mathcal{NP}$-hard, we observed that there is a special sub-class of problems with 'well-chosen' matrices, that is solvable with

polynomial-time algorithms. This might have a great impact on system design for future devices using Compressed Sensing techniques. It is also likely that sparse modeling will become a large field in digital signal processing and other fields of applied mathematics, since this is the key for Compressed Sensing techniques to work.

Appendix

A Least Squares Solutions and the Moore-Penrose Pseudoinverse

Overdetermined Systems of Linear Equations

Consider the overdetermined system of linear equations

$$\mathbf{y} = \mathbf{A}\,\mathbf{x}\,, \tag{A.1}$$

where $\mathbf{A}$ is of size $M \times N$ with $N < M$, $\mathrm{rank}(\mathbf{A}) = N$, $\mathbf{y} \in \mathbb{C}^M$, and the unknown $\mathbf{x} \in \mathbb{C}^N$. The least squares approach aims to find

$$\hat{\mathbf{x}} = \underset{\mathbf{x}\in\mathbb{C}^N}{\mathrm{argmin}} \, \|\mathbf{y} - \mathbf{A}\,\mathbf{x}\|_2^2 \,. \tag{A.2}$$

To minimize the cost function

$$\begin{aligned} J(\mathbf{x}) \;&:=\; \|\mathbf{y} - \mathbf{A}\,\mathbf{x}\|_2^2 \;=\; (\mathbf{y} - \mathbf{A}\,\mathbf{x})^{\mathrm{H}}\,(\mathbf{y} - \mathbf{A}\,\mathbf{x}) && \text{(A.3)} \\ &=\; \mathbf{y}^{\mathrm{H}}\mathbf{y} \;-\; \mathbf{y}^{\mathrm{H}}\mathbf{A}\,\mathbf{x} \;-\; \mathbf{x}^{\mathrm{H}}\mathbf{A}^{\mathrm{H}}\,\mathbf{y} \;+\; \mathbf{x}^{\mathrm{H}}\mathbf{A}^{\mathrm{H}}\mathbf{A}\,\mathbf{x} && \text{(A.4)} \end{aligned}$$

we set the gradient

$$\frac{\partial\, J(\mathbf{x})}{\partial\, \mathbf{x}^*} \;=\; -\,\mathbf{A}^{\mathrm{H}}\mathbf{y} \;+\; \mathbf{A}^{\mathrm{H}}\mathbf{A}\mathbf{x} \tag{A.5}$$

to zero. This leads to the system of normal equations

$$\mathbf{A}^{\mathrm{H}}\,\mathbf{y} \;=\; \mathbf{A}^{\mathrm{H}}\mathbf{A}\,\mathbf{x}\,, \tag{A.6}$$

and, since $\mathrm{rank}(\mathbf{A}) = N$, to the explicit expression

$$\hat{\mathbf{x}} \;=\; \left(\mathbf{A}^{\mathrm{H}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathrm{H}}\,\mathbf{y}\,. \tag{A.7}$$

Defining

$$\mathbf{A}^{+} \;:=\; \left(\mathbf{A}^{\mathrm{H}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathrm{H}}\,, \tag{A.8}$$

the least squares solution of (A.1) reads

$$\hat{\mathbf{x}} \;=\; \mathbf{A}^{+}\,\mathbf{y}\,. \tag{A.9}$$

Underdetermined Systems of Linear Equations

Now consider

$$\mathbf{y} = \mathbf{A}\,\mathbf{x}$$

to be an *under*determined system with $\mathbf{A} \in \mathbb{C}^{M\times N}$ and $M<N$, and let $\text{rank}(\mathbf{A}) = M$. Now the matrix $\mathbf{A}^{\text{H}}\mathbf{A}$ is not invertible and the above approach using the normal equations (A.6) is not applicable. Therefore we must choose another way. Using the LQ factorization

$$\mathbf{A} = [\,\mathbf{L} \mid 0\,]\,\mathbf{Q} \tag{A.10}$$

with unitary $\mathbf{Q} \in \mathbb{C}^{N\times N}$ and lower left triangular $\mathbf{L} \in \mathbb{C}^{M\times M}$, the system reads

$$\mathbf{y} = \mathbf{A}\,\mathbf{x} = [\,\mathbf{L} \mid 0\,]\,\mathbf{Q}\,\mathbf{x}\,. \tag{A.11}$$

Defining

$$\mathbf{z} = \mathbf{Q}\,\mathbf{x} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix} \tag{A.12}$$

with $\mathbf{z}_1 \in \mathbb{C}^M$ and $\mathbf{z}_2 \in \mathbb{C}^{N-M}$, it follows

$$\mathbf{y} = [\,\mathbf{L} \mid 0\,]\,\mathbf{z} = \mathbf{L}\,\mathbf{z}_1\,, \tag{A.13}$$

which has a unique solution for $\mathbf{z}_1$. Hence, any $\mathbf{z} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{bmatrix}$ with $\mathbf{z}_1 = \mathbf{L}^{-1}\,\mathbf{y}$ and arbitrary $\mathbf{z}_2 \in \mathbb{C}^{N-M}$ is a solution of $\mathbf{y} = [\,\mathbf{L} \mid 0\,]\,\mathbf{z}$, which can be inserted in A.12 to solve for $\mathbf{x}$. Since $\mathbf{Q}$ is unitary, the set of solutions is determined by all

$$\mathbf{x} \in \left\{ \mathbf{Q}^{\text{H}} \begin{bmatrix} \mathbf{L}^{-1}\,\mathbf{y} \\ \mathbf{z}_2 \end{bmatrix} \;\middle|\; \mathbf{z}_2 \in \mathbb{C}^{N-M} \right\}. \tag{A.14}$$

To obtain the least squares solution which minimizes $\|\mathbf{x}\|_2$ we choose $\mathbf{z}_2 = \mathbf{0}$, yielding

$$\begin{aligned} \hat{\mathbf{x}} &= \mathbf{Q}^{\text{H}} \begin{bmatrix} \mathbf{L}^{-1}\,\mathbf{y} \\ \mathbf{0} \end{bmatrix} \\ &= \mathbf{Q}^{\text{H}} \begin{bmatrix} \mathbf{L}^{\text{H}} \left(\mathbf{L}^{\text{H}}\right)^{-1} \mathbf{L}^{-1}\,\mathbf{y} \\ \mathbf{0} \end{bmatrix} \\ &= \mathbf{Q}^{\text{H}} \begin{bmatrix} \mathbf{L}^{\text{H}} \\ \mathbf{0} \end{bmatrix} \left(\mathbf{L}\,\mathbf{L}^{\text{H}}\right)^{-1} \mathbf{y} \\ &= \mathbf{A}^{\text{H}} \left(\mathbf{A}\,\mathbf{A}^{\text{H}}\right)^{-1} \mathbf{y}\,. \end{aligned} \tag{A.15}$$

Defining

$$\mathbf{A}^{+} := \mathbf{A}^{\mathrm{H}} \left(\mathbf{A}\,\mathbf{A}^{\mathrm{H}}\right)^{-1}, \tag{A.16}$$

we obtain the closed form expression of the least squares solution

$$\hat{\mathbf{x}} = \mathbf{A}^{+}\,\mathbf{y}. \tag{A.17}$$

The Moore-Penrose Pseudoinverse

Definition A.1. Let $\mathbf{A} \in \mathbb{C}^{M\times N}$. The *Moore-Penrose pseudoinverse* of $\mathbf{A}$ is the uniquely determined matrix $\mathbf{A}^{+} \in \mathbb{C}^{N\times M}$ which satisfies the *Moore-Penrose conditions*:

$$\text{(i)} \quad \mathbf{A}\,\mathbf{A}^{+}\mathbf{A} = \mathbf{A} \tag{A.18}$$

$$\text{(ii)} \quad \mathbf{A}^{+}\mathbf{A}\,\mathbf{A}^{+} = \mathbf{A}^{+} \tag{A.19}$$

$$\text{(iii)} \quad \left(\mathbf{A}\,\mathbf{A}^{+}\right)^{\mathrm{H}} = \mathbf{A}\,\mathbf{A}^{+} \tag{A.20}$$

$$\text{(iv)} \quad \left(\mathbf{A}^{+}\mathbf{A}\right)^{\mathrm{H}} = \mathbf{A}^{+}\mathbf{A} \tag{A.21}$$

By verifying the Moore-Penrose conditions it can be shown that $\mathbf{A}^{+}$ as defined in (A.8) for $M > N$, and in (A.16) for $M < N$ (both with full rank) is the unique Moore-Penrose pseudoinverse in both cases.

B Some Variants of the Noisy ℓ_1 - Minimization Problem

Besides the ℓ_1-relaxation (P_1') to (P_0'), there are other problem formulations involving the ℓ_1-norm to solve the sparse estimation problem. A few of those approaches are presented in the following. The examples are taken from [5], where some further examples are listed. Also, in the literature one can find many more problem formulations which are, to some degree, adapted to specific applications.

Lagrangian Form

The Lagrangian form involves the Lagrange multiplier $\lambda > 0$, which is dependent on the noise bound ε (introduced at page 13):

$$\text{(B.1)} \qquad \text{Minimize} \quad \|\mathbf{x}\|_1 + \frac{\lambda}{2} \|\mathbf{y} - \mathbf{\Phi}\,\mathbf{x}\|_2^2$$

For a specific choice of λ this formulation is equivalent to (P_1'). However, the exact relation between λ and ε is unknown.

LASSO Formulation

The LASSO formulation depends on a parameter τ and flips the objective and constraint:

$$\text{(B.2)} \qquad \begin{aligned} &\text{Minimize} && \frac{1}{2} \|\mathbf{y} - \mathbf{\Phi}\,\mathbf{x}\|_2^2 \\ &\text{subject to} && \|\mathbf{x}\|_1 \leq \tau \end{aligned}$$

Again, there is a relation between τ and ε, and again the exact relation is unknown.

Dantzig Selector

The Dantzig Selector [20] uses a different approach and involves some infinity norm constraint with a parameter $\delta \geq 0$, which results from duality considerations:

$$\begin{aligned} &\text{Minimize} && \|\mathbf{x}\|_1 \\ &\text{subject to} && \left\|\boldsymbol{\Phi}^{\mathsf{H}}(\mathbf{y} - \boldsymbol{\Phi}\,\mathbf{x})\right\|_\infty \leq \delta \end{aligned} \tag{B.3}$$

In the noiseless case, i.e. for $\varepsilon = 0$, this problem is equivalent to (P_1') and (P_1) for the choice of $\delta = 0$ (provided that $\boldsymbol{\Phi}$ has maximal rank).

C Wirtinger's Calculus

For derivation of complex-valued functions one can make use of some results due to Wilhelm Wirtinger in [85]. We consider only a simple implication of the original work, which significantly simplifies calculations, see [57].

Let $f : \mathbb{C} \to \mathbb{C}$ and $z \in \mathbb{C}$. We can write $z = x + \mathrm{i}\, y$, $x, y \in \mathbb{R}$, and interpret f as a function $\mathbb{R}^2 \to \mathbb{R}^2$ which we require to be differentiable in the real-valued sense. Then the following equations hold

$$\frac{\partial f}{\partial z} = \frac{1}{2}\left(\frac{\partial f}{\partial x} - \mathrm{i}\frac{\partial f}{\partial y}\right) \tag{C.1}$$

$$\frac{\partial f}{\partial z^*} = \frac{1}{2}\left(\frac{\partial f}{\partial x} + \mathrm{i}\frac{\partial f}{\partial y}\right) . \tag{C.2}$$

Simple calculations lead to the important implications that

$$\frac{\partial z^*}{\partial z} = 0 \quad \text{and} \quad \frac{\partial z}{\partial z^*} = 0 . \tag{C.3}$$

That means that when deriving f with respect to z^* we can treat z as a constant and vice versa.

Example C.1. Let $f : \mathbb{C} \to \mathbb{R}$ with $z \mapsto |z|^2$. This function is not complex differentiable since it does not satisfy the Cauchy-Riemann equations. But we can write

$$f(z) = z\, z^* = (x + \mathrm{i}\, y)(x - \mathrm{i}\, y) = x^2 + y^2 ,$$

which is differentiable in the real-valued sense. Using Wirtinger's calculus it follows

$$\frac{\partial f}{\partial z} = z^* \quad \text{and} \quad \frac{\partial f}{\partial z^*} = z .$$

If for instance we are interested in a minimum of $f(z)$ we have the necessary condition

$$\frac{\partial}{\partial z^*} f(z) \stackrel{!}{=} 0 ,$$

which leads to $z = 0$ as a possible solution.

This concept can be generalized for complex-valued vector analysis. In analogy to equation (C.3), for some vector $\mathbf{x} \in \mathbb{C}^N$ it holds

$$\frac{\partial \mathbf{x}^*}{\partial \mathbf{x}} = \frac{\partial \mathbf{x}^\mathsf{H}}{\partial \mathbf{x}} = \mathbf{0} \quad \text{and} \quad \frac{\partial \mathbf{x}}{\partial \mathbf{x}^*} = \mathbf{0} \,. \tag{C.4}$$

Example C.2. Let $f : \mathbb{C}^N \to \mathbb{R}$ with $\mathbf{x} \mapsto \|\mathbf{A}\,\mathbf{x}\|_2^2$ for some matrix $\mathbf{A}$ with rank N. We can write

$$f(\mathbf{x}) \;=\; (\mathbf{A}\,\mathbf{x})^\mathsf{H}\,\mathbf{A}\,\mathbf{x} \;=\; \mathbf{x}^\mathsf{H}(\mathbf{A}^\mathsf{H}\mathbf{A})\,\mathbf{x} \;=\; \mathbf{x}^\mathsf{H}\mathbf{Q}\,\mathbf{x}\,,$$

where $\mathbf{Q} = \mathbf{A}^\mathsf{H}\mathbf{A}$ is conjugate symmetric and positive definite.

If we are interested in the $\mathbf{x}$ that minimizes the quadratic functional f, we can calculate the gradient with respect to $\mathbf{x}^*$, that is

$$\frac{\partial f}{\partial \mathbf{x}^*} = \mathbf{Q}\,\mathbf{x}\,,$$

and set it to zero. In this small example this would yield $\mathbf{x} = \mathbf{0}$, since $\mathbf{Q}$ is quadratic and has full rank.

List of Figures

List of Tables

List of Algorithms

Nomenclature

Abbreviations

AIHT	Accelerated Iterative Hard Thresholding
AWGN	Additive white Gaussian noise
BP	Basis Pursuit
CoSaMP	Compressed Sampling Matching Pursuit
DCT	Discrete Cosine transform
DFT	Discrete Fourier transform
DORE	Double Over Relaxation
GP	Gradient Pursuit
i.i.d.	independent and identically distributed
IDFT	Inverse discrete Fourier transform
IHT	Iterative Hard Thresholding
ISD	Iterative Support Detection
MP	Matching Pursuit
MRI	Magnetic resonance imaging
NSE	Normalized squared error
NSP	Null Space Property
OFDM	Orthogonal frequency-division multiplexing
OMP	Orthogonal Matching Pursuit
RIP	Restricted Isometry Property
SL0	Smoothed ℓ_0 algorithm
SNR	Signal-to-noise ratio
SOCP	Second-order cone program

Symbols and Operators

$(\cdot)^*$	Complex conjugate of a scalar, matrix or vector
$(\cdot)^{\mathsf{H}}$	Complex conjugate transposition of a matrix or vector
$(\cdot)^{-1}$	Inverse matrix or operator
$(\cdot)^{+}$	Pseudoinverse of a matrix
$(\cdot)^{\mathsf{T}}$	Transposition of a matrix or vector
$\mathrm{Re}(z)$	Real part of $z \in \mathbb{C}$
$\mathrm{Im}(z)$	Imaginary part of $z \in \mathbb{C}$
$\mathbf{0}$	The null vector
$\mathbf{e}_k$	Canonical unit vector with entries $(\mathbf{e}_k)_l = \delta_{k,l}$
$\delta_{k,l}$	Kronecker delta, $\delta_{k,l} = 1$ if $k = l$ and $\delta_{k,l} = 0$ otherwise
δ_S	Isometry constant of order S
$\mathbf{I}$	Identity matrix
i	The imaginary unit $\mathrm{i} := \sqrt{-1}$
$\ker(\mathbf{A})$	Kernel / null space of a matrix $\mathbf{A}$
$\|\mathbf{x}\|_0$	Number of non-zero entries in $\mathbf{x}$
$\|\mathbf{x}\|_p$	ℓ_p-norm or quasinorm of a vector $\mathbf{x}$, $p > 0$
$\mathbf{A}^{(\mathcal{C})}$	Column sub-matrix of a matrix $\mathbf{A}$, indexed by a set $\mathcal{C}$
$\mathbf{A}_{(\mathcal{R})}$	Row sub-matrix of a matrix $\mathbf{A}$, indexed by a set $\mathcal{R}$
$\mathbf{A}_{(\mathcal{R})}^{(\mathcal{C})}$	Sub-matrix of a matrix $\mathbf{A}$, consisting of the rows and columns, indexed by a $\mathcal{R}$ and $\mathcal{C}$, respectively
$\mathbf{F}$	Fourier (DFT) matrix
$\mu(\boldsymbol{\Phi}, \boldsymbol{\Psi})$	Coherence between two bases $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$
$\mu(\boldsymbol{\Phi})$	Mutual coherence of a dictionary or matrix $\boldsymbol{\Phi}$
$\mathrm{rank}(\mathbf{A})$	Rank of a matrix $\mathbf{A}$, i.e. the maximal number of linearly independent rows or columns
$\mathrm{spark}(\mathbf{A})$	Spark of a matrix $\mathbf{A}$, i.e. the minimal number of linearly dependent columns
$\mathrm{supp}(\mathbf{x})$	Support set of a vector $\mathbf{x}$, i.e. the set of indices where $\mathbf{x}$ has non-zero entries
$\varnothing$	The empty set
$B_p(R)$	ℓ_p-ball of radius R
$\mathrm{H}_S(\mathbf{x})$	Hard thresholding operator, setting all but the S largest entries of $\mathbf{x}$ to zero
$\sigma_S(\mathbf{x})_p$	Best S-term approximation error

Bibliography

[1] F. Alizadeh and D. Goldfarb. Second-Order Cone Programming. *Mathematical Programming*, 95:3–51, 2001.

[2] R. G. Baraniuk. Compressive Sensing. *Lecture Notes in IEEE Signal Processing Magazine*, 24(4):118–120, Jul. 2007.

[3] R. G. Baraniuk, M. Davenport, R. DeVore, and M. Wakin. A Simple Proof of the Restricted Isometry Property for Random Matrices. *Constructive Approximation*, 28:253–263, 2008.

[4] D. Baron, M. B. Wakin, M. F. Duarte, S. Sarvotham, and R. G. Baraniuk. *Distributed Compressed Sensing*, Nov. 2006.

[5] S. Becker. Sparse- and low-rank approximation wiki, Category:Problems. online. `http://www.ugcs.caltech.edu/~srbecker/wiki/Category:Problems`.

[6] T. Blumensath. Software package sparsify. online. `http://users.fmrib.ox.ac.uk/~tblumens/sparsify/sparsify.html`.

[7] T. Blumensath. Accelerated iterative hard thresholding. *Signal Processing*, 92(3):752–756, 2012.

[8] T. Blumensath and M. E. Davies. Gradient Pursuits. *Signal Processing, IEEE Transactions on*, 56(6):2370–2382, June 2008.

[9] T. Blumensath and M. E. Davies. A simple, efficient and near optimal algorithm for compressed sensing. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 3357–3360, April 2009.

[10] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[11] E. Candès and J. Romberg. ℓ_1-MAGIC: Recovery of Sparse Signals via Convex Programming. 4, 2005. `http://www-stat.stanford.edu/~candes/l1magic/downloads/l1magic.pdf`.

[12] E. J. Candès. Compressive sampling. In *Proceedings of the International Congress of Mathematicians: Madrid, August 22-30, 2006: invited lectures*, pages 1433–1452, 2006.

[13] E. J. Candès. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9-10):589–592, 2008.

[14] E. J. Candès and J. Romberg. Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23(3):969, 2007.

[15] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *Information Theory, IEEE Transactions on*, 52(2):489–509, Feb. 2006.

[16] E. J. Candès, J. Romberg, and T. Tao. Stable Signal Recovery from Incomplete and Inaccurate Measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.

[17] E. J. Candès, M. Rudelson, T. Tao, and R. Vershynin. Error Correction via Linear Programming. In *Foundations of Computer Science, 2005. FOCS 2005. 46th Annual IEEE Symposium on*, pages 668–681, Oct. 2005.

[18] E. J. Candès and T. Tao. Decoding by Linear Programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, Dec. 2005.

[19] E. J. Candès and T. Tao. Near-Optimal Signal Recovery From Random Projections: Universal Encoding Strategies? *Information Theory, IEEE Transactions on*, 52(12):5406–5425, Dec. 2006.

[20] E. J. Candès and T. Tao. The Dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics*, 35(6):2313–2351, 2007.

[21] E. J. Candès and M. B. Wakin. An Introduction To Compressive Sampling. *Signal Processing Magazine, IEEE*, 25(2):21–30, March 2008.

[22] R. Chartrand. Exact Reconstruction of Sparse Signals via Nonconvex Minimization. *IEEE Signal Process. Lett.*, 14:707–710, 2007.

[23] R. Chartrand. Nonconvex Compressed Sensing and Error Correction. In *32nd International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2007.

[24] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic Decomposition by Basis Pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

[25] C. Christopoulos, A. Skodras, and T. Ebrahimi. The JPEG2000 still image coding system: an overview. *Consumer Electronics, IEEE Transactions on*, 46(4):1103–1127, Nov 2000.

[26] A. Cohen, W. Dahmen, and R. DeVore. Compressed sensing and best k-term approximation. *J. Amer. Math. Soc*, 22(1):211–231, 2009.

[27] R. Coifman, F. Geshwind, and Y. Meyer. Noiselets. *Applied and Computational Harmonic Analysis*, 10(1):27–44, 2001.

[28] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd edition, 2001.

[29] I. CVX Research. CVX: Matlab Software for Disciplined Convex Programming, version 2.0 beta. `http://cvxr.com/cvx`, Sept. 2012.

[30] W. Dai and O. Milenkovic. Subspace Pursuit for Compressive Sensing Signal Reconstruction. *Information Theory, IEEE Transactions on*, 55(5):2230–2249, May 2009.

[31] M. A. Davenport. *Random Observations on Random Observations: Sparse Signal Acquisition and Processing*. PhD thesis, Rice University, Aug. 2010.

[32] M. A. Davenport and M. B. Wakin. Analysis of Orthogonal Matching Pursuit Using the Restricted Isometry Property. *Information Theory, IEEE Transactions on*, 56(9):4395–4401, Sept. 2010.

[33] M. E. Davies and T. Blumensath. Faster & Greedier: algorithms for sparse reconstruction of large datasets. In *Communications, Control and Signal Processing, 2008. ISCCSP 2008. 3rd International Symposium on*, pages 774–779, March 2008.

[34] R. A. DeVore. Nonlinear approximation. *Acta Numerica*, pages 51–150, 1998.

[35] R. A. DeVore. Deterministic constructions of compressed sensing matrices. *Journal of Complexity*, 23:918–925, 2007.

[36] D. Donoho. Compressed sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, April 2006.

[37] D. Donoho, M. Elad, and V. Temlyakov. Stable Recovery of Sparse Overcomplete Representations in the Presence of Noise. *Information Theory, IEEE Transactions on*, 52(1):6–18, Jan. 2006.

[38] D. Donoho and J. Tanner. Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1906):4273–4293, 2009.

[39] D. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck. Sparse Solution of Underdetermined Systems of Linear Equations by Stagewise Orthogonal Matching Pursuit. *Information Theory, IEEE Transactions on*, 58(2):1094–1121, Feb. 2012.

[40] D. L. Donoho. For Most Large Underdetermined Systems of Linear Equations the Minimal ℓ^1-norm Solution is also the Sparsest Solution. *Comm. Pure Appl. Math*,

59:797–829, 2004.

[41] D. L. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization. *Proceedings of the National Academy of Sciences*, 100(5):2197–2202, 2003.

[42] D. L. Donoho and X. Huo. Uncertainty Principles and Ideal Atomic Decomposition. *Information Theory, IEEE Transactions on*, 47(7):2845–2862, Nov. 2001.

[43] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk. Single-Pixel Imaging via Compressive Sampling. *Signal Processing Magazine, IEEE*, 25(2):83–91, March 2008.

[44] D. Eiwen, G. Tauböck, F. Hlawatsch, and H. Feichtinger. Compressive tracking of doubly selective channels in multicarrier systems based on sequential delay-Doppler sparsity. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 2928–2931, May 2011.

[45] M. Elad and A. M. Bruckstein. A Generalized Uncertainty Principle and Sparse Representation in Pairs of Bases. *Information Theory, IEEE Transactions on*, 48(9):2558–2567, Sept. 2002.

[46] Y. Eldar and G. Kutyniok. *Compressed Sensing: Theory and Applications*. Cambridge University Press, 2012.

[47] M. Fornasier and H. Rauhut. Compressive Sensing. In O. Scherzer, editor, *Handbook of Mathematical Methods in Imaging*, pages 187–228. Springer, 2011.

[48] S. Foucart. Hard Thresholding Pursuit: An Algorithm for Compressive Sensing. *SIAM Journal on Numerical Analysis*, 49(6):2543–2563, 2011.

[49] M. Gay. Compressed Channel Sensing – First Results. Technical report, Fraunhofer IIS, 2012.

[50] M. Gay. Strategy and Implementation Considerations for Sparse OFDM Channel Estimation. Technical report, Fraunhofer IIS, 2012.

[51] M. Gay, A. Lampe, and M. Breiling. Sparse OFDM Channel Estimation Based on Regular Pilot Grids. In *9th International ITG Conference on Systems, Communications and Coding 2013 (SCC'2013)*, Munich, Germany, Jan. 2013.

[52] S. A. Geršgorin. Über die Abgrenzung der Eigenwerte einer Matrix. *Izv. Akad. Nauk SSSR Ser. Fiz.-Mat.*, (6):749–754, 1931.

[53] G. Golub and C. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 1996.

[54] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs.

In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. http://stanford.edu/~boyd/graph_dcp.html.

[55] J. Haupt and R. Nowak. Signal Reconstruction From Noisy Random Projections. *Information Theory, IEEE Transactions on*, 52(9):4036–4048, Sept. 2006.

[56] P. Indyk. Explicit Constructions for Compressed Sensing of Sparse Signals. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2008.

[57] K. D. Kammeyer. *Nachrichtenübertragung*. Vieweg + Teubner, 2008.

[58] D. E. Knuth. *The Art of Computer Programming, Volume 3 (2nd ed.): Sorting and Searching*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 1998.

[59] J. Kovačević and A. Chebira. Life Beyond Bases: The Advent of Frames (Part I). *Signal Processing Magazine, IEEE*, 24(4):86–104, July 2007.

[60] J. Kovačević and A. Chebira. Life Beyond Bases: The Advent of Frames (Part II). *Signal Processing Magazine, IEEE*, 24(5):115–125, Sept. 2007.

[61] G. Kutyniok. Compressed Sensing: Theory and Applications. *arXiv preprint arXiv:1203.3815*, 2012.

[62] E. Livshitz. On efficiency of Orthogonal Matching Pursuit. *arXiv preprint arXiv:1004.3946*, 2010.

[63] H. D. Lüke. The Origins of the Sampling Theorem. *Communications Magazine, IEEE*, 37(4):106–108, April 1999.

[64] A. Maleki. Coherence Analysis of Iterative Thresholding Algorithms. In *Communication, Control, and Computing, 47th Annual Allerton Conference on. Allerton 2009.*, pages 236–243, 30 2009-Oct. 2 2009.

[65] S. G. Mallat and Z. Zhang. Matching Pursuits with Time-Frequency Dictionaries. *Signal Processing, IEEE Transactions on*, 41(12):3397–3415, Dec. 1993.

[66] H. Mohimani, M. Babaie-Zadeh, and C. Jutten. Complex-valued sparse representation based on smoothed ℓ^0 norm. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 3881–3884, 31 2008-April 4 2008.

[67] H. Mohimani, M. Babaie-Zadeh, and C. Jutten. A fast approach for overcomplete sparse decomposition based on smoothed ℓ^0 norm. *Signal Processing, IEEE Transactions on*, 57(1):289–301, 2009.

[68] B. Natarajan. Sparse Approximate Solutions to Linear Systems. *SIAM Journal on Computing*, 24(2):227–234, 1995.

[69] D. Needell and J. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.

[70] D. Needell, J. Tropp, and R. Vershynin. Greedy Signal Recovery Review. In *Signals, Systems and Computers, 2008 42nd Asilomar Conference on*, pages 1048–1050, Oct. 2008.

[71] D. Needell and R. Vershynin. Uniform Uncertainty Principle and Signal Recovery via Regularized Orthogonal Matching Pursuit. *Foundations of computational mathematics*, 9(3):317–334, 2009.

[72] D. Needell and R. Vershynin. Signal Recovery From Incomplete and Inaccurate Measurements Via Regularized Orthogonal Matching Pursuit. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):310–316, April 2010.

[73] Y. Pati, R. Rezaiifar, and P. Krishnaprasad. Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition. In *Signals, Systems and Computers, Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44 vol.1, Nov. 1993.

[74] W. Pennebaker and J. Mitchell. JPEG Still Image Data Compression Standard. *New York: Van Nostrand Reinhold*, 1, 1993.

[75] K. Qiu and A. Dogandzic. ECME Thresholding Methods for Sparse Signal Reconstruction. *arXiv*, (1004.4880v3), 2010.

[76] M. Rudelson and R. Vershynin. On sparse reconstruction from Fourier and Gaussian measurements. *Communications on Pure and Applied Mathematics*, 61(8):1025–1045, 2008.

[77] C. E. Shannon. Communication in the Presence of Noise. *Proceedings of the IRE*, 37(1):10–21, Jan. 1949.

[78] J. A. Tropp. Greed is Good: Algorithmic Results for Sparse Approximation. *Information Theory, IEEE Transactions on*, 50(10):2231–2242, Oct. 2004.

[79] J. A. Tropp and A. C. Gilbert. Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit. *Information Theory, IEEE Transactions on*, 53(12):4655–4666, Dec. 2007.

[80] J. A. Tropp and S. J. Wright. Computational Methods for Sparse Solution of Linear Inverse Problems. *Proceedings of the IEEE*, 98(6):948–958, June 2010.

[81] D. Tse and P. Viswanath. *Fundamentals Of Wireless Communication*. Cambridge University Press, 2005.

[82] G. K. Wallace. The JPEG still picture compression standard. *Consumer Electronics, IEEE Transactions on*, 38(1):xviii–xxxiv, Feb. 1992.

[83] Y. Wang and W. Yin. Sparse Signal Reconstruction via Iterative Support Detection. *SIAM Journal on Imaging Sciences*, 3(3):462–491, 2010.

[84] L. Welch. Lower Bounds on the Maximum Cross Correlation of Signals (Corresp.). *Information Theory, IEEE Transactions on*, 20(3):397–399, May 1974.

[85] W. Wirtinger. Zur formalen Theorie der Funktionen von mehr komplexen Veränderlichen. *Math. Ann.*, 97:357–375, 1926.

[86] T. Zhang. Sparse Recovery with Orthogonal Matching Pursuit under RIP. *arXiv*, abs/1005.2249, 2010.

Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Mittweida, 28. Februar 2013 _____

Unterschrift